

# Global Practical Synchronization in Kuramoto Networks: A Submodular Optimization Framework

Andrew Clark, *Member, IEEE*, Basel Alomair, *Member, IEEE*, Linda Bushnell, *Senior Member, IEEE* and Radha Poovendran, *Fellow, IEEE*

**Abstract**—Synchronization underlies phenomena including memory and perception in the brain, coordinated motion of animal flocks, and stability of the power grid. These synchronization phenomena are often modeled through networks of phase-coupled oscillating nodes. Heterogeneity in the node dynamics, however, may prevent such networks from achieving the required level of synchronization. In order to guarantee synchronization, external inputs can be used to pin a subset of nodes to a reference frequency, while the remaining nodes are steered toward synchronization via local coupling. In this paper, we present a submodular optimization framework for selecting a set of nodes to act as external inputs in order to achieve synchronization from almost any initial network state. We derive threshold-based sufficient conditions for synchronization, and then prove that these conditions are equivalent to connectivity of a class of augmented network graphs. Based on this connection, we map the sufficient conditions for synchronization to constraints on submodular functions, leading to efficient algorithms with provable optimality bounds for selecting input nodes. We illustrate our approach via numerical studies of synchronization in networks from power systems, wireless networks, and neuronal networks.

## I. INTRODUCTION

Synchronization plays a vital role in complex networks. Stable operation of the power grid requires synchronization of buses and generators to a common frequency [1]. Synchronized oscillations of neuronal firing provide a biological mechanism for aggregating information in perception [2] and memory [3]. Coordinated motion of animals [4] occurs when a common heading is achieved. The prevalence of synchronization across different application domains motivates the study of the basic principles underlying synchronization [5].

Phase-coupled oscillators have been proposed as a widely-applicable framework for studying synchronization [6]. In phase-coupled oscillator networks, the dynamics of each node's phase are determined by a diffusive coupling with its neighbors, together with an intrinsic frequency. In particular, the Kuramoto model [7], which assumes sinusoidal coupling, has been extensively studied and is related to synchronization phenomena in power grids. The sinusoidal coupling causes the node phases to approach synchronization, while the intrinsic frequency drives each node away from synchronization.

The existence and stability of synchronized states has been studied extensively in the literature [8], [9], [10], including conditions based on the intrinsic frequencies, network topology, and degree of coupling between the nodes. An important case is synchronization in the presence of external inputs [11]. External inputs arise in applications including neuroscience, where they represent environmental stimuli [2] or deep brain stimulation [12]. From an engineering standpoint, by introducing external inputs that pin a subset of nodes to a desired phase and frequency, a network that does not synchronize in the absence of inputs can be driven to a synchronized state, thus facilitating stability and performance of the network.

Existing analytical approaches to introducing external inputs assume that the external input node is connected to all other nodes [13], or that the network has a specific topology such as a complete graph [14]. Developing sufficient conditions for synchronization using external inputs in networks with arbitrary topology is an open problem. Efficient algorithms for selecting a subset of input nodes in order to guarantee synchronization are also not available in the existing literature.

In this paper, we present an optimization framework for selecting a subset of phase-coupled oscillators to act as external inputs in order to guarantee synchronization of a Kuramoto network. We formulate sufficient conditions for ensuring convergence of phase-coupled oscillators to a synchronized state from any given set of initial states. We then develop a submodular optimization framework for selecting a minimum-size set of input nodes to achieve a desired level of synchronization, with provable bounds on the optimality of the chosen input nodes. Our contributions are summarized as follows:

- We investigate two synchronization problems. The first problem is to ensure that the phases of all of the oscillators converge to fixed points that are within a desired bound of a given reference phase (practical node synchronization). The second problem is to guarantee that the phase differences of neighboring nodes converge to within a desired bound of each other (practical edge synchronization). In both problems, the node frequencies must converge to the same value.
- We derive a set of sufficient conditions for a given set of input nodes to achieve practical node or edge synchronization from any given set of initial states. We interpret our conditions as each node achieving a desired level of synchronization if a threshold number of neighbors reaches that level of synchronization.
- We develop a submodular optimization framework for

A. Clark is with the Department of Electrical and Computer Engineering, Worcester Polytechnic Institute, Worcester, MA, 01609. [aclark@wpi.edu](mailto:aclark@wpi.edu)

L. Bushnell and R. Poovendran are with the Department of Electrical Engineering, University of Washington, Seattle, WA, 98195-2500. {lb2, rp3}@uw.edu

B. Alomair is with the Center for Cybersecurity, King Abdulaziz City for Science and Technology, Riyadh, Saudi Arabia. [alomair@kacst.edu.sa](mailto:alomair@kacst.edu.sa)

selecting a minimum-size set of input nodes to achieve these synchronization conditions. Our approach is to derive a connection between the threshold conditions and the connectivity of an augmented graph. Based on this connection, we map our sufficient conditions to constraints on submodular functions. We propose efficient algorithms for selecting input nodes to guarantee synchronization and analyze the optimality bounds of our algorithms.

- We evaluate our approach through a numerical study of synchronization in three classes of networks. First, we consider synchronization of power grids using the IEEE 14 Bus test case [15]. Second, we study synchronization in geometric random graphs, motivated by vehicle coordination and wireless network synchronization problems. Finally, we investigate synchronization of neuronal networks based on the *C. Elegans* dataset [16].

The rest of the paper is organized as follows. In Section II, we give an overview of the related work. Section III contains our system model and definitions of synchronization. Sufficient conditions for synchronization are presented in Section IV. In Section V, we describe our submodular optimization approach to selecting input nodes that satisfy the sufficient conditions. Section VI contains our numerical study. In Section VII, we conclude the paper and discuss directions for future work.

## II. RELATED WORK

The phase-coupled oscillator framework for modeling synchronization phenomena was introduced in the seminal work of Winfree [17]. Models of oscillation include Lorenz [18], kick [19], Van der Pol [20], and pulse-coupled [21] oscillators. The sinusoidally-coupled Kuramoto model was introduced in [22]. Extensive studies have been performed on the mean-field behavior of the Kuramoto model with all-to-all coupling (i.e., each node is coupled to each other node) in the limit as the network size grows large [7].

Synchronization of the Kuramoto model with a finite number of oscillators and arbitrary connection topology was studied in [9]. The authors proved that convergence to the synchronized state is guaranteed when all nodes have identical intrinsic frequencies, and analyzed the feasibility and stability of synchronization under non-identical intrinsic frequencies. Stable equilibria of the Kuramoto model in finite networks, including both synchronized and non-synchronized states, were analyzed in [23]. More recently, the existence, uniqueness, and stability of partially synchronized states was studied in [24], with application to power networks [1]. These prior works considered synchronization in the absence of external inputs.

Synchronization in the presence of external inputs has achieved relatively less study. Numerical studies have estimated the region of attraction, defined as the set of initial states that converge to the desired state, of the Kuramoto model with inputs for the case of all-to-all coupling [25]. Sufficient conditions for synchronization when there is a single input node that is connected to all other nodes were presented in [13], [26], [14]. These works do not, however, consider synchronization with external inputs in networks with arbitrary

topology, and do not propose methods for selecting a subset of input nodes.

Steering a complex network to a desired state by pinning a set of nodes to a fixed value has been studied in the area of pinning control [27], [28], [29], [30]. These existing works assume that each node's dynamics is a nonlinear function of the current state with linear coupling between the neighbors, and hence differs from our approach that considers nonlinear coupling between neighboring nodes. To the best of our knowledge, conditions for pinning controllability of the Kuramoto model that we consider are not available in the existing literature.

Selecting external input nodes to achieve synchronization can be viewed as part of the broader area of selecting input nodes to control complex networks. Much recent work on selecting input nodes has focused on guaranteeing controllability of linear dynamics on the network [31], [32]. The assumption of linear dynamics, however, is not applicable to the nonlinear oscillators considered here.

Phase-coupled oscillators in general, and Kuramoto oscillators in particular, have found extensive applications. Coupled oscillators were initially used to model natural phenomena such as bird flocking [33] and fish schooling [4]. At the level of individual cells, phase-coupled oscillators also provide a framework for heart pacemakers [34] and neuronal networks [3]. The prevalence of phase-coupled oscillators in nature has inspired engineering techniques for formation control [4] and time synchronization [35]. The phases of buses and generators in the power grid have also been modeled using phase-coupled oscillators, in order to understand whether the grid maintains a required level of synchronization for stability [36].

## III. MODEL AND PRELIMINARIES

In this section, we describe the system model and oscillator dynamics. We then define the notions of synchronization considered in this paper. Finally, we give a preliminary result that will be needed for the proof of Theorem 2 in Section IV.

### A. System Model

A network of  $n$  oscillators, indexed in the set  $V = \{1, \dots, n\}$  is considered (we use the term *oscillator* and *node* interchangeably, as the oscillators correspond to nodes in a graph). Each oscillator  $v \in V$  has a neighbor set  $N(v) \subseteq V$ , consisting of the set of oscillators that are coupled to  $v$ . We assume that links are bidirectional, so that  $u \in N(v)$  implies  $v \in N(u)$ . An edge  $(u, v)$  exists if  $u \in N(v)$  and  $v \in N(u)$ . We let  $E$  denote the set of edges. The graph  $G = (V, E)$  is assumed to be connected; if not, our proposed input selection methods can be applied to each connected component of the graph.

Each oscillator  $v$  has a time-varying phase  $\theta_v(t)$ . The vector of oscillator phases at time  $t$  is denoted  $\boldsymbol{\theta}(t) \in \mathbb{R}^n$ . We assume that there are two types of oscillators, denoted *input* and *non-input* oscillators. We let  $A$  denote the set of input oscillators. The phases of the non-input oscillators follow the Kuramoto dynamics [7]

$$\dot{\theta}_v(t) = - \sum_{u \in N(v)} K_{uv} \sin(\theta_v(t) - \theta_u(t)) + \omega_v. \quad (1)$$

In (1), the first term represents the coupling between the oscillators, while  $\omega_v$  is the intrinsic frequency of oscillator  $v$  and describes the phase dynamics in the absence of coupling. The coupling coefficient  $K_{uv} > 0$  determines the relative strength of the two terms. We assume throughout that the couplings between nodes are symmetric, so that  $K_{uv} = K_{vu}$  for all  $(u, v) \in E$ .

Each input oscillator  $v \in A$  is assumed to be pinned to a desired frequency  $\omega_0$  and phase offset  $\theta_0$ , so that  $\dot{\theta}_v(t) = \omega_0$  and  $\theta_v(t) = \omega_0 t + \theta_0$  for all  $v \in A$ . The overall oscillator dynamics are given by

$$\dot{\theta}_v(t) = \begin{cases} -\sum_{u \in N(v)} K_{uv} \sin(\theta_v(t) - \theta_u(t)) + \omega_v, & v \notin A \\ \omega_0, & v \in A \end{cases} \quad (2)$$

We define a function  $\sigma(x) : [-2\pi, 2\pi] \rightarrow [-\pi, \pi]$  as

$$\sigma(x) = \begin{cases} x + 2\pi, & x \in [-2\pi, -\pi) \\ x, & x \in [-\pi, \pi) \\ x - 2\pi, & x \in [\pi, 2\pi] \end{cases}$$

The function  $f(x)$  maps elements in  $[-2\pi, 2\pi]$  into  $[-\pi, \pi]$  while satisfying  $\sin(\sigma(x)) = \sin(x)$  and  $\cos(\sigma(x)) = \cos(x)$  for all  $x$ . This function will be used to define the edge cohesiveness property in the following subsection.

### B. Definitions of Synchronization

We now define the notions of synchronization considered in this paper. Analogous definitions for networks without external inputs are given in [24].

The strongest form of synchronization is *phase synchronization*, defined as follows.

**Definition 1:** The oscillators achieve *phase synchronization* if there exists  $\theta^*$  such that  $\lim_{t \rightarrow \infty} \theta_v(t) = \theta^*$  for all  $v \in V$ .

Phase synchronization is achieved if all oscillators converge to the same phase. Since synchronization of all oscillators to the same phase is not possible in general [9], weaker notions of synchronization have been proposed. We first define *node-* and *edge-cohesiveness* as follows.

**Definition 2 (Node Cohesiveness):** A network of oscillators achieves  $\gamma$ -*node cohesiveness* with parameter  $\gamma \in [0, \frac{\pi}{4}]$  if there exists  $T > 0$  such that, for all  $t \geq T$ ,  $|\theta_v(t) - (\omega_0 t + \theta_0)| < \gamma$ .

Node cohesiveness is achieved if all nodes converge to within a desired bound of the input node phase  $(\omega_0 t + \theta_0)$ . The requirement that  $\gamma \in [0, \frac{\pi}{4}]$  ensures that the relative phase differences between nodes are bounded by  $\pi/2$ , which is a requirement of synchronization in many real-world systems [1], [24].

**Definition 3 (Edge Cohesiveness):** A network of oscillators achieves  $\gamma$ -*edge cohesiveness* with parameter  $\gamma \in [0, \frac{\pi}{2}]$  if there exists  $T > 0$  such that, for all  $t \geq T$  and all  $(u, v) \in E$ ,  $|\sigma(\theta_v - \theta_u)| < \gamma$ .

Edge cohesiveness implies that the relative differences between any pair of neighboring nodes is bounded above by  $\gamma$ . In general, node cohesiveness is desirable in applications such as coordinated motion [4] and time synchronization [35], where all nodes must agree on a common phase. Edge cohesiveness is desirable in applications including power systems [1], where

the relative differences between nodes must be within a certain range to ensure stability.

We observe that a network that is  $\gamma$ -node cohesive is  $(2\gamma)$ -edge cohesive. Indeed, node cohesiveness implies that there exists  $T > 0$  such that, for all  $t \geq T$ ,

$$\begin{aligned} |\theta_v(t) - \theta_u(t)| &= |(\theta_v(t) - (\omega_0 t + \theta_0)) - (\theta_u(t) - (\omega_0 t + \theta_0))| \\ &\leq |\theta_v(t) - (\omega_0 t + \theta_0)| + |\theta_u(t) - (\omega_0 t + \theta_0)| \\ &< \gamma + \gamma = 2\gamma. \end{aligned}$$

An additional synchronization notion is frequency synchronization, defined as follows.

**Definition 4 (Frequency Synchronization):** The oscillators achieve *frequency synchronization* if there exists  $\omega^*$  such that  $\lim_{t \rightarrow \infty} \dot{\theta}_v(t) = \omega^*$  for all  $v \in V$ .

Note that the only possible value for  $\omega^*$  in Definition 4 is the frequency of the input nodes, denoted  $\omega_0$ , since we have  $\dot{\theta}_v(t) \equiv \omega_0$  for all  $v \in A$  and  $t > 0$ .

We now define the main types of synchronization considered in this paper.

**Definition 5 (Practical Node and Edge Synchronization):** The oscillators achieve  $\gamma$ -*practical node synchronization* if they achieve frequency synchronization and  $\gamma$ -node cohesiveness. The oscillators achieve  $\gamma$ -*practical edge synchronization* if they achieve frequency synchronization and  $\gamma$ -edge cohesiveness.

The following lemma allows us to focus on the case where  $\omega_0 = \theta_0 = 0$ , so that all input oscillators have frequency and phase 0.

**Lemma 1:** Define  $\hat{\theta}(t) = \theta(t) - (\omega_0 t + \theta_0)\mathbf{1}$ , where  $\mathbf{1}$  denotes the vector of all 1's. Then  $\theta(t)$  achieves  $\gamma$ -practical node (resp. edge) synchronization with frequency  $\omega_0$  and input node phase  $\theta_0$  if and only if  $\hat{\theta}(t)$  achieves  $\gamma$ -practical node (resp. edge) synchronization with frequency 0 and input node phase 0.

**Proof:** First, note that  $\dot{\hat{\theta}}_v(t) = \dot{\theta}_v(t) - \omega_0$  for all  $v \in V$ . Suppose that  $\theta(t)$  achieves  $\gamma$ -practical node synchronization with frequency  $\omega_0$  and reference phase  $\theta_0$ . Then for all  $v \in V$ ,

$$\lim_{t \rightarrow \infty} \dot{\hat{\theta}}_v(t) = \lim_{t \rightarrow \infty} (\dot{\theta}_v(t) - \omega_0) = 0.$$

Furthermore, there exists  $T > 0$  such that, for all  $t \geq T$  and  $v \in V$ ,

$$|\hat{\theta}_v(t)| = |\theta_v(t) - (\theta_0 + \omega_0 t)| \leq \gamma,$$

and hence  $\gamma$ -practical node synchronization is achieved.

Now, if  $\theta(t)$  achieves  $\gamma$ -practical edge synchronization with frequency  $\omega_0$  and reference phase  $\theta_0$ , then there exists  $T > 0$  such that, for all  $t \geq T$  and  $v \in V$ ,

$$|\sigma(\hat{\theta}_v - \hat{\theta}_u)| = |\sigma(\theta_v - \theta_u)| \leq \gamma.$$

The proof of the converse is similar. ■

For  $\gamma \in [0, \frac{\pi}{4}]$ , define  $\Lambda_{final} = \{|\theta|_\infty \leq \gamma\}$ . Let  $\Lambda_{init}$  be a set of feasible initial states, defined by  $\Lambda_{init} = \{\theta : |\theta_i| \leq \theta_i^0\}$  for some  $\theta_1^0, \dots, \theta_n^0$ . Finally, let  $\Lambda_{bound}$  be defined by  $\Lambda_{bound} = \{\theta : |\theta_i| \leq \theta_i^{max}\}$  for some  $\theta_1^{max}, \dots, \theta_n^{max}$ . The desired condition that we consider for a set of input nodes to guarantee practical synchronization is defined as follows.

**Definition 6:** Let  $\Lambda_{final}$  and  $\Lambda_{init}$  denote subsets of  $[-\pi, \pi]^n$  with  $\Lambda_{init} \subseteq \Lambda_{bound}$  and  $\Lambda_{final} \cap \Lambda_{bound} \neq \emptyset$ . A set of input nodes is said to *guarantee practical synchronization* if for any  $\theta(0) \in \Lambda_{init}$ , (i)  $\lim_{t \rightarrow \infty} \theta(t) = \theta^*$  for some  $\theta^* \in \Lambda_{final}$ , and (ii)  $\theta(t) \in \Lambda_{bound}$  for all  $t \geq 0$ .

In Definition 6, the condition (i) implies that the oscillators eventually achieve frequency synchronization at an equilibrium point that lies within a desired region  $\Lambda_{final}$ . Condition (ii) implies that, prior to convergence, the oscillators remain within a desired “safe” region  $\Lambda_{bound}$ . As an example, in a power system, the goal may be to ensure that a set of generators reach frequency synchronization with relative phase differences between the nodes within a desired region  $\Lambda_{final}$ , while ensuring that

$$\theta(t) \in \Lambda_{bound} \triangleq \left\{ \theta : |\theta_v - \theta_u| \leq \frac{\pi}{2} \quad \forall (u, v) \in E \right\}$$

for all  $t \geq 0$ . If this condition does not hold, generators will lose synchronism with respect to each other, and one or more generators may trip to avoid hardware damage, potentially leading to grid instability.

Finally, we define a metric that describes the worst-case distance of each oscillator from the input node phase at equilibrium.

**Definition 7:** Suppose that a set of oscillators achieves frequency synchronization for any  $\theta(0) \in \Lambda_{init}$ . The *input cohesiveness* is defined by

$$g(A) = \max \{ \|\theta^*\|_\infty : \theta^* \in \Theta \} \quad (3)$$

where

$$\Theta = \{ \theta \in \mathbb{R}^n : \lim_{t \rightarrow \infty} \theta(t) = \theta^* \text{ for some } \theta(0) \in \Lambda_{init} \}.$$

The metric  $g(A)$  corresponds to the smallest  $\gamma$  such that the oscillators achieve  $\gamma$ -practical node synchronization from input set  $A$ .

### C. Preliminary Result

The following preliminary result of [37] will be needed in Section IV. First, for any  $\delta > 0$  and any matrix  $B$ , the  $\delta$ -digraph is defined as the digraph where edge  $(i, j)$  exists if  $B_{ij} \geq \delta$ .

**Theorem 1 ([37], Theorem 1):** Consider the linear system  $\dot{x}(t) = F(t)x(t)$ , where  $F(t)$  is a time-varying system matrix. Assume that the system matrix is a bounded piecewise continuous function of time, and that for every time  $t$  the system matrix is Metzler (i.e., all off-diagonal elements are nonnegative) and has zero row sums. Suppose further that there is an index  $k \in \{1, \dots, n\}$ , a threshold value  $\delta > 0$ , and an interval length  $T > 0$  such that for all  $t \in \mathbb{R}$  the  $\delta$ -digraph associated to

$$\int_t^{t+T} F(s) ds$$

has the property that all nodes may be reached from the node  $k$ . Then the set of states  $\{x^* \mathbf{1} : x^* \in \mathbb{R}\}$  is uniformly exponentially stable. In particular, all components of any solution converge to a common value as  $t \rightarrow \infty$ .

The theorem gives a sufficient condition for a linear system with time-varying weights to converge to consensus. Theorem

1 will be used to prove that, under certain conditions on the oscillator phases, the frequencies  $\{\dot{\theta}_v(t) : v \in V\}$  converge to the frequencies of the input nodes.

## IV. SUFFICIENT CONDITIONS FOR PRACTICAL SYNCHRONIZATION

In this section, we formulate the problem of selecting a set of inputs to guarantee practical synchronization and provide sufficient conditions for a set of inputs to achieve practical synchronization. We first show that the oscillators achieve practical synchronization if there exists a positive invariant set containing the set of possible initial states of the oscillators, and if any initial state in the invariant set will eventually converge to the set of admissible final states. We then derive sufficient conditions for existence of the positive invariant set, as well as convergence to practical synchronization.

### A. Statement of Sufficient Condition

The following theorem gives a sufficient condition for a set of oscillators to guarantee practical synchronization.

**Theorem 2:** Suppose that there exists a set  $\Lambda_{PI}$  such that the following conditions hold: (a)  $\Lambda_{init} \subseteq \Lambda_{PI} \subseteq \Lambda_{bound}$ , (b)  $\Lambda_{PI}$  is *positive invariant*, i.e., if  $\theta(0) \in \Lambda_{PI}$ , then  $\theta(t) \in \Lambda_{PI}$  for all  $t \geq 0$ , and (c) If  $\theta(0) \in \Lambda_{PI}$ , then there exists  $T$  such that  $t \geq T$  implies  $\theta(t) \in \Lambda_{final}$ . Then the set of input nodes  $A$  satisfies conditions (i) and (ii) of Definition 6 and hence guarantees practical synchronization.

**Proof:** In order to prove condition (i) of Definition 6, consider  $\dot{\theta}(t)$ , which has dynamics

$$\ddot{\theta}_v(t) = - \sum_{u \in N(v)} \left[ K_{uv} \cos(\theta_v(t) - \theta_u(t)) (\dot{\theta}_v(t) - \dot{\theta}_u(t)) \right] \quad (4)$$

for  $v \notin A$  and  $\ddot{\theta}_v(t) \equiv 0$  for  $v \in A$ . We now define dynamics of the form in Theorem 1 in order to analyze the convergence of (4). Let  $x(t) \in \mathbb{R}^{n+1}$  denote the state variable, where  $x_{n+1}$  is the state of a “super node” with dynamics  $\dot{x}_{n+1}(t) \equiv 0$ . Define the system matrix  $F(t)$  by

$$F_{vu}(t) = \begin{cases} \cos K_{uv}(\theta_v(t) - \theta_u(t)), & \text{for } (u, v) \in E, v \notin A \\ -\sum_{s \in N(v)} K_{sv} \cos(\theta_v(t) - \theta_s(t)), & u = v, u \notin A \\ -1, & \text{for } u = v, u \in A \\ 1, & \text{for } v \in A, u = (n+1) \\ 0, & \text{for } v = n+1 \end{cases}$$

By condition (c), there exists  $T > 0$  such that  $|\theta_v(t) - \theta_u(t)| < \pi/2$  for all  $t > T$ . Hence  $F(t)$  is a bounded, piecewise continuous Metzler matrix with rows that sum to zero, and the connectivity of the graph  $G$  implies that node  $(n+1)$  is connected to all other nodes in the associated  $\delta$ -digraph. By Theorem 1,  $x(t)$  converges to a state  $x^* \mathbf{1}$ .

Now, if we set  $x_v(0) = x_{n+1}(0) = 0$  for all  $v \in A$  and  $x_v(0) = \dot{\theta}_v(0)$  for all  $v \notin A$ , then the trajectory of  $[x_1(t) \cdots x_n(t)]^T$  will be identical to the trajectory of  $\dot{\theta}(t)$ . Hence, by Theorem 1,  $\lim_{t \rightarrow \infty} \dot{\theta}(t) = \omega^* \mathbf{1}$  for some  $\omega^* \in \mathbb{R}$ . Moreover, since  $\dot{\theta}_v(t) \equiv 0$  for all  $v \in A$ , we must have  $\omega^* = 0$ , implying frequency synchronization.

Since  $\dot{\theta}$  converges to 0,  $\theta(t)$  converges to a fixed point  $\theta^*$ . By condition (c), since  $\theta(t) \in \Lambda_{final}$  for  $t$  sufficiently large,

$\theta^* \in \Lambda_{final}$ . This completes the proof of condition (i) of Definition 6.

The proof of condition (ii) in Definition 6 follows from (a) and (b), which imply that  $\theta(t) \in \Lambda_{PI} \subseteq \Lambda_{bound}$  for all  $t \geq 0$  whenever  $\theta(0) \in \Lambda_{init}$ . ■

Figure 1 illustrates the relationship between the sets  $\Lambda_{bound}$ ,  $\Lambda_{init}$ ,  $\Lambda_{final}$ , and  $\Lambda_{PI}$  of Definition 6 and Theorem 2.

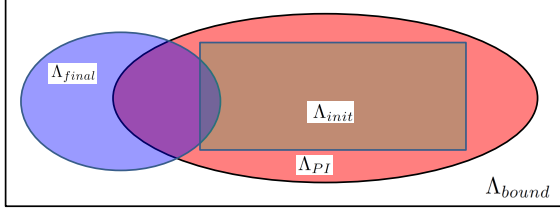


Fig. 1. Illustration of the sets  $\Lambda_{bound}$ ,  $\Lambda_{init}$ ,  $\Lambda_{final}$ , and  $\Lambda_{PI}$ .

We observe that, if conditions (a) and (b) of Theorem 2 hold, then the oscillators achieve frequency synchronization and the input cohesiveness metric  $g(A)$  is well-defined. The following section describes a procedure for testing whether a positive invariant set  $\Lambda_{PI}$  satisfying the conditions (a)–(c) of Theorem 2 exists for a given set of input nodes  $A$ .

### B. Identifying a Positive Invariant Set

In this section, we develop a procedure for identifying a subset  $\Lambda_{PI}$  that is positive invariant and satisfies  $\Lambda_{init} \subseteq \Lambda_{PI} \subseteq \Lambda_{bound}$ , for a given set of input nodes  $A$ , provided that such a positive invariant set exists. This corresponds to conditions (a) and (b) of Theorem 2. We first state the following proposition that gives a sufficient condition for positive invariance.

**Proposition 1:** Suppose that a set  $\Lambda_{PI}$  is defined by  $\Lambda_{PI} = \{\theta : |\theta_v| \leq \bar{\theta}_v\}$  for some  $\{\bar{\theta}_v : v \in V\}$ . If

$$\sum_{u \in N(v)} \min \{K_{uv} \sin(\bar{\theta}_v - \theta_u) : |\theta_u| \leq \bar{\theta}_u\} > |\omega_v| + \epsilon \quad (5)$$

for all  $v \in V$  and some  $\epsilon > 0$ , then the set  $\Lambda_{PI}$  is positive invariant.

*Proof:* Suppose that (5) holds for all  $v \in V$  and yet  $\Lambda_{PI}$  is not positive invariant. Let  $\theta(0) \in \Lambda_{PI}$  be such that  $\theta(t) \notin \Lambda_{PI}$  for some  $t$ , and define

$$t^* \triangleq \inf \{t : \theta_v(t) > \bar{\theta}_v \text{ for some } v\}.$$

Then  $|\theta_v(t^*)| = \bar{\theta}_v$  for some  $v \in V$ , and we have

$$\begin{aligned} \dot{\theta}_v(t^*) &= - \sum_{u \in N(v)} K_{uv} \sin(\bar{\theta}_v - \theta_u(t^*)) + \omega_v \\ &< - \sum_{u \in N(v)} \min \{K_{uv} \sin(\bar{\theta}_v - \theta_u) : |\theta_u| \leq \bar{\theta}_u\} + |\omega_v| \\ &< -(|\omega_v| + \epsilon) + |\omega_v| < -\epsilon \end{aligned}$$

implying that  $|\theta_v(t)| \leq \bar{\theta}_v$  in some neighborhood of  $t^*$  and contradicting the definition of  $t^*$ .

A similar argument holds when  $\theta_v(t^*) = -\bar{\theta}_v$ . We have

$$\begin{aligned} \dot{\theta}_v(t^*) &= - \sum_{u \in N(v)} K_{uv} \sin(-\bar{\theta}_v - \theta_u(t^*)) + \omega_v \\ &= \sum_{u \in N(v)} K_{uv} \sin(\bar{\theta}_v + \theta_u(t^*)) + \omega_v \\ &> \sum_{u \in N(v)} \min \{K_{uv} \sin(\bar{\theta}_v - \theta_u) : |\theta_u| \leq \bar{\theta}_u\} + \omega_v \\ &> |\omega_v| + \epsilon - |\omega_v| > \epsilon \end{aligned}$$

which implies that  $\theta_v(t) > -\bar{\theta}_v$  in some neighborhood of  $t^*$ .

These contradictions imply that  $\Lambda_{PI}$  is positive invariant. ■

Proposition 1 can be interpreted as, when any oscillator  $v$  reaches the boundary of  $\Lambda_{PI}$ , the coupling with the neighboring oscillators is sufficient to draw  $v$  back to the interior of  $\Lambda_{PI}$ . The proposition leads to the following procedure for computing a positive invariant set  $\Lambda_{PI}$ , if such a set exists, for a given set of input nodes  $A$ . Select an integer  $M > 0$ , and define  $\mathbf{X}[k]$  to be a vector in  $\{0, \dots, M\}^n$ , where  $k$  is a discrete time index. Let  $r(v) = \lceil \frac{M\theta_v^0}{\theta_v^{max}} \rceil$ , and initialize  $\mathbf{X}[0]$  as  $X_v[0] = r(v)$  for  $v \notin A$  and  $X_v[0] = 0$  for  $v \in A$ . At the  $k$ -th iteration of the algorithm, select  $v \in V$  such that

$$\sum_{u \in N(v)} \min \left\{ K_{uv} \sin \left( \frac{\theta_v^{max} X_v[k-1]}{M} - \theta_u \right) : |\theta_u| \leq \frac{\theta_u^{max} X_u[k-1]}{M} \right\} \leq |\omega_v| + \epsilon. \quad (6)$$

Set  $X_v[k] = X_v[k-1] + 1$ , and  $X_u[k] = X_u[k-1]$  for  $u \neq v$ . The algorithm terminates when  $X_v[k] = M$  for some  $v \in V$ , or when no index  $v$  satisfies the conditions of (6). A pseudocode description is given as Algorithm 1.

The algorithm discretizes the interval  $[0, \gamma]$  by dividing it into  $M$  intervals. Hence, a larger value of  $M$  results in more granular intervals and a more precise approximation of the positive invariant set.

The following theorem describes the guarantees provided by Algorithm 1.

**Theorem 3:** Suppose that Algorithm 1 returns a nonempty vector  $(\bar{\theta}_1, \dots, \bar{\theta}_n)$ . The set  $\Lambda_{PI} = \{\theta : |\theta_v| \leq \bar{\theta}_v\}$  is positive invariant and satisfies  $\Lambda_{init} \subseteq \Lambda_{PI} \subseteq \Lambda_{bound}$ .

*Proof:* The approach is to show that  $\Lambda_{PI}$  satisfies the conditions of Proposition 1. Suppose that there exists  $v$  such that (5) does not hold. Then

$$\sum_{u \in N(v)} \min \{K_{uv} \sin(\bar{\theta}_v - \theta_u) : |\theta_u| \leq \bar{\theta}_u\} \leq |\omega_v| + \epsilon,$$

and hence Algorithm 1 would increment the value of  $X_v[k]$  instead of returning  $(\bar{\theta}_1, \dots, \bar{\theta}_n)$ . This contradiction implies that  $\Lambda_{PI}$  is positive invariant.

We now show that  $\Lambda_{init} \subseteq \Lambda_{PI} \subseteq \Lambda_{bound}$ . For each  $v \notin A$ ,

$$\bar{\theta}_v = \frac{\theta_v^{max} X_v[k]}{M} \geq \frac{\theta_v^{max} r(v)}{M} \geq \theta_v^0,$$

**Algorithm 1** Algorithm for computing a positive invariant set  $\Lambda_{PI}$  associated with input nodes  $A$ .

---

```

1: procedure IDENTIFY_PI_SET( $G = (V, E)$ ,  $\omega_1, \dots, \omega_n$ ,
    $A$ ,  $\theta_v^0 : v \in V$ ,  $\theta_v^{max} : v \in V$ )
2:   Input: Graph  $G = (V, E)$ , intrinsic frequencies
    $\omega_1, \dots, \omega_n$ , input nodes  $A$ , initial states  $\{\theta_v^0 : v \in V\}$ ,
   boundary values  $\{\theta_v^{max} : v \in V\}$ 
3:   Output:  $(\bar{\theta}_1, \dots, \bar{\theta}_n)$  such that  $\{\theta : |\theta_v| \leq \bar{\theta}_v\}$  is
   positive invariant, or  $\emptyset$  if no such set can be found
4:   Initialization:  $r(v) \leftarrow \lceil \frac{\theta_v^0 M}{\theta_v^{max}} \rceil$ ,  $X_v[k] \leftarrow 0$  for  $v \in A$ ,
    $X_v[k] \leftarrow r(v)$  for  $v \notin A$ ,  $k \leftarrow 0$ 
5:   while 1 do
6:      $k \leftarrow k + 1$ ,  $\mathbf{X}[k] \leftarrow \mathbf{X}[k - 1]$ 
7:     for  $v \notin A$  do
8:        $\rho_{uv} \leftarrow \min \left\{ \sin \left( \frac{\theta_v^{max} X_v[k-1]}{M} - \theta_u \right) : \right.$ 
    $|\theta_u| \leq \frac{\theta_u^{max} X_u[k-1]}{M} \left. \right\}$ 
9:        $\rho_v \leftarrow \sum_{u \in N(v)} K_{uv} \rho_{uv}$ 
10:    end for
11:    if  $\rho_v > |\omega_v| + \epsilon \forall v$  then
12:      return  $\left( \frac{\theta_1^{max} X_1[k-1]}{M}, \dots, \frac{\theta_n^{max} X_n[k-1]}{M} \right)$ 
13:    else
14:      Choose  $v$  such that  $\rho_v \leq |\omega_v| + \epsilon$ 
15:       $X_v[k] \leftarrow X_v[k - 1] + 1$ 
16:      if  $X_v[k] == M$  then return  $\emptyset$  end if
17:    end if
18:  end while
19: end procedure

```

---

implying that  $\Lambda_{init} \subseteq \Lambda_{PI}$ . Furthermore, if Algorithm 1 does not return  $\emptyset$ , then  $X_v[k] < M$  when the algorithm terminates and hence  $\bar{\theta}_v < \theta_v^{max}$ , implying that  $\Lambda_{PI} \subseteq \Lambda_{bound}$ . ■

Algorithm 1 provides an efficient procedure for verifying existence of a positive invariant set  $\Lambda_{PI}$  and constructing such a set for a given set of input nodes  $A$ . In addition, this procedure will form the basis for the submodular optimization approach to selecting input nodes presented in Section V-A.

### C. Verifying Convergence to $\Lambda_{final}$

The following proposition gives sufficient conditions to ensure that, for any  $\theta(0) \in \Lambda_{init}$ , there exists  $T$  such that  $\theta(t) \in \Lambda_{final}$  for all  $t \geq T$ .

**Proposition 2:** Let  $\Lambda = \{\theta : |\theta_v| \leq \bar{\theta}_v\}$  be a positive invariant set. Suppose there exists an index  $v$  and  $\underline{\theta}_v \in [0, \bar{\theta}_v]$  such that

$$\sum_{u \in N(v)} \min \{K_{uv} \sin(\theta_v - \theta_u) : \theta_v \in [\underline{\theta}_v, \bar{\theta}_v], |\theta_u| \leq \bar{\theta}_u\} > |\omega_v| + \epsilon. \quad (7)$$

Define  $\Lambda' = \Lambda \setminus \{\theta : \theta_v \in [\underline{\theta}_v, \bar{\theta}_v]\}$ . Then  $\Lambda'$  is positive invariant and, when  $\theta(0) \in \Lambda$ , there exists  $T$  such that  $\theta(t) \in \Lambda'$  for all  $t \geq T$ .

**Proof:** We first show positive invariance of  $\Lambda'$ . Suppose  $\theta(0) \in \Lambda'$ . Since  $\Lambda$  is positive invariant,  $|\theta_u| \leq \bar{\theta}_u$  for all  $u \neq v$ . It suffices to show that if  $|\theta_v(0)| \leq \underline{\theta}_v$ , then  $|\theta_v(t)| \leq \underline{\theta}_v$  for all  $t \geq 0$ . This result holds by (7) and Proposition 1.

It remains to prove that, if  $\theta(0) \in \Lambda$ , then there exists  $T$  such that  $\theta(t) \in \Lambda'$  for  $t \geq T$ . Suppose that this is not the case, and choose  $\theta(0) \in \Lambda$  such that  $\theta_v(t) \in [\underline{\theta}_v, \bar{\theta}_v]$  for all  $t \geq 0$ . Then

$$\begin{aligned} \dot{\theta}_v(t) &= - \sum_{u \in N(v)} K_{uv} \sin(\theta_v(t) - \theta_u(t)) + \omega_v \\ &\leq -|\omega_v| - \epsilon + |\omega_v| = -\epsilon, \end{aligned}$$

implying that

$$\theta_v(t) \leq \theta_v(0) - \epsilon t \leq \bar{\theta}_v - \epsilon t$$

and hence  $\theta_v(t) \leq \underline{\theta}_v$  for  $t$  sufficiently large. This contradiction implies that  $\theta(T) \in \Lambda'$  for some  $T > 0$ . Positive invariance of  $\Lambda'$  then yields  $\theta(t) \in \Lambda'$  for  $t \geq T$ . ■

We now present an efficient algorithm for checking whether, for any  $\theta(0) \in \Lambda_{init}$ , there exists  $T \geq 0$  such that  $\theta(t) \in \Lambda_{final}$  for all  $t \geq T$ . We assume that Algorithm 1 has already been used to construct a set  $\Lambda_{PI}$ , satisfying the conditions (a) and (b) of Theorem 2, of the form  $\Lambda_{PI} = \{\theta : |\theta_v| \leq \theta_v^{PI}\}$ .

Let  $M$  be defined as in Section IV-B. For each  $v$ , let  $r(v) = \lfloor \frac{M\gamma}{\theta_v^{max}} \rfloor$  and  $s(v) = \lceil \frac{M\theta_v^{PI}}{\theta_v^{max}} \rceil$ . Define a vector  $\mathbf{X}[k] \in \{0, \dots, M\}^n$ , where  $k$  is a time index. Define  $X_v[0] = s(v)$  for  $v \notin A$  and  $X_v[0] = 0$  for  $v \in A$ . At time step  $k$ , select an index  $v$  such that

$$\begin{aligned} \sum_{u \in N(v)} \min \{K_{uv} \sin(\theta_v - \theta_u) : \\ \theta_v \in \left[ \frac{(X_v[k-1] - 1)\theta_v^{max}}{M}, \frac{X_v[k-1]\theta_v^{max}}{M} \right], \\ |\theta_u| \leq \frac{X_u[k-1]\theta_u^{max}}{M} \} > |\omega_v| + \epsilon \quad (8) \end{aligned}$$

if such an index exists. Set  $X_v[k] = X_v[k-1] - 1$  and  $X_u[k] = X_u[k-1]$  if  $u \neq v$ . The algorithm terminates if no index  $v$  satisfying (8) can be found. A pseudocode description of the algorithm is given as Algorithm 2.

The following theorem describes how the results of Algorithm 2 can be used to verify convergence to  $\Lambda_{final}$ .

**Theorem 4:** Suppose that there exists a positive invariant set  $\Lambda_{PI}$ , and that  $\bar{\theta} = (\bar{\theta}_1, \dots, \bar{\theta}_n)$  is returned by Algorithm 2. Let  $\bar{\Lambda} = \{\theta : |\theta_v| \leq \bar{\theta}_v\}$ . If  $\bar{\Lambda} \subseteq \Lambda_{final}$ , then for each  $\theta(0) \in \Lambda_{init}$ , there exists  $T$  such that  $\theta(t) \in \Lambda_{final}$  for all  $t \geq T$ .

**Proof:** Let  $\Lambda_k = \left\{ \theta : |\theta_v| \leq \frac{X[k]\theta_v^{max}}{M} \right\}$ . The approach of the proof is to show that for all  $k$ ,  $\theta(0) \in \Lambda_{PI}$  implies that there exists  $T$  with  $\theta(t) \in \Lambda_k$  for  $t \geq T$ .

The proof is by induction on  $k$ . When  $k = 0$ ,  $\Lambda_0 = \Lambda_{PI}$ , and hence the result holds by the positive invariance of  $\Lambda_{PI}$ . At step  $k$ , the index  $v$  selected by the algorithm satisfies (8), and hence Proposition 2 holds with  $\Lambda = \Lambda_{k-1}$ ,  $\bar{\theta}_v = \frac{X_v[k-1]\theta_v^{max}}{M}$ , and  $\underline{\theta}_v = \frac{(X_v[k-1]-1)\theta_v^{max}}{M}$ . In this case,  $\Lambda_{k-1} \setminus \{\theta : \theta_v \in [\underline{\theta}_v, \bar{\theta}_v]\} = \Lambda_k$ . By Proposition 2,  $\theta(t) \in \Lambda_k$  for  $t$  sufficiently large, establishing the inductive step.

The fact that  $\theta(t) \in \Lambda_{final}$  for  $t$  sufficiently large then follows from  $\bar{\Lambda} \subseteq \Lambda_{final}$ . ■

If Algorithm 2 returns **False**, then the set of input nodes  $A$  is insufficient to guarantee convergence to  $\Lambda_{final}$ , and hence

**Algorithm 2** Algorithm for proving that a set of input nodes  $A$  is sufficient to ensure convergence to  $\Lambda_{final}$ .

```

1: procedure CHECK_CONVERGENCE( $G = (V, E)$ ,
    $\omega_1, \dots, \omega_n$ ,  $A$ ,  $\theta_v^0 : v \in V$ ,  $\theta_v^{max} : v \in V$ ,  $\theta_v^{PI} : v \in V$ ,
    $\theta_v^{PI} : v \in V$ )
2:   Input: Graph  $G = (V, E)$ , intrinsic frequencies
    $\omega_1, \dots, \omega_n$ , input nodes  $A$ , initial states  $\{\theta_v^0 : v \in V\}$ ,
   boundary values  $\{\theta_v^{max} : v \in V\}$ , positive invariant
   set defined by  $\{\theta_v^{PI} : v \in V\}$ , set  $\Lambda_{final}$  defined by
    $\{\theta_v' : v \in V\}$ .
3:   Output: Return true if, for any  $\theta \in \Lambda_{PI}$ ,  $\theta(t) \in \Lambda_{final}$ 
   for  $t$  sufficiently large. Return false otherwise.
4:   Initialization:  $r(v) \leftarrow \lfloor \frac{M\theta_v^{PI}}{\theta_v^{max}} \rfloor$ ,  $s(v) \leftarrow \lceil \frac{M\theta_v^{PI}}{\theta_v^{max}} \rceil$ ,  $k \leftarrow 0$ ,
    $X_v[k] \leftarrow 0$  for  $v \in A$ ,  $X_v[k] \leftarrow s(v)$  for  $v \notin A$ 
5:   while 1 do
6:      $k \leftarrow k + 1$ ,  $\mathbf{X}[k] \leftarrow \mathbf{X}[k - 1]$ 
7:     for  $v \notin A$  do
8:        $\underline{\theta}_v \leftarrow \frac{(X_v[k-1] - 1)\theta_v^{max}}{M}$ 
9:        $\bar{\theta}_v \leftarrow \frac{X_v[k-1]\theta_v^{max}}{M}$ 
10:      for  $u \in N(v)$  do
11:         $\bar{\theta}_u \leftarrow \frac{X_u[k-1]\theta_u^{max}}{M}$ 
12:         $\rho_{uv} \leftarrow \min \{ \sin(\theta_v - \theta_u) : \theta_v \in [\underline{\theta}_v, \bar{\theta}_v], \theta_u \in [\bar{\theta}_u, \theta_u^{max}] \}$ 
13:      end for
14:       $\rho_v \leftarrow \sum_{u \in N(v)} K_{uv} \rho_{uv}$ 
15:    end for
16:    if  $\rho_v \leq |\omega_v| + \epsilon \forall v$  then
17:       $\bar{\theta}_v \leftarrow \frac{\theta_v^{max} X_v[k-1]}{M} \forall v \in V$ 
18:      if  $\bar{\theta}_v \leq \theta_v'$  for all  $v \in V$  then
19:        return true
20:      else
21:        return false
22:      end if
23:    else
24:      Choose  $v$  such that  $\rho_v > |\omega_v| + \epsilon$ 
25:       $X_v[k] \leftarrow X_v[k - 1] - 1$ 
26:    end if
27:  end while
28: end procedure

```

a new input set must be selected. Selecting such a set is the topic of Section V.

#### D. Example

Consider a line network, with  $V = \{n_1, n_2, n_3\}$  and  $E = \{(n_1, n_2), (n_2, n_3)\}$ , in which the set of input nodes is given by  $A = \{n_1\}$ . In the network,  $K_{uv} = 1$  for all  $(i, j) \in E$ , and set  $\omega_1 = 0.0705$ ,  $\omega_2 = 0.0709$ , and  $\omega_3 = 0.0336$ . The set  $\Lambda_{bound} = \{\|\theta\|_\infty \leq \frac{\pi}{4}\}$ ,  $\Lambda_{init} = \{\|\theta\|_\infty \leq \frac{\pi}{5}\}$ , and  $\Lambda_{final} = \{\|\theta\|_\infty \leq \frac{\pi}{6}\}$ .

Set  $M = 20$  and consider Algorithm 1. With  $M = 20$ ,  $\mathbf{X}[0] = (0, 16, 16)$ . At the first iteration,  $X_3$  is incremented, resulting in  $\mathbf{X}[1] = (0, 16, 17)$ . The algorithm terminates, and hence  $\Lambda_{PI}$  is defined by  $\bar{\theta} = \{0, \frac{\pi}{5}, \frac{17\pi}{80}\}$ .

Applying Algorithm 2 with these values of  $\Lambda_{PI}$  proceeds as follows. This time,  $\mathbf{X}[0] = (0, 16, 17)$ . The values of  $\mathbf{X}$  at subsequent iterations are shown in Figure 2.

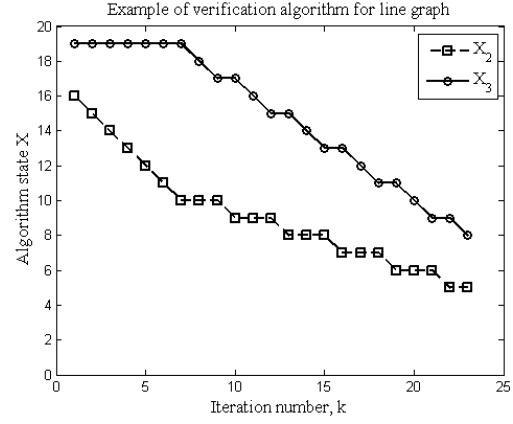


Fig. 2. Values of  $X_2$  and  $X_3$  at each iteration of Algorithm 2 for a line graph ( $n_1$  is the input and hence  $X_1[k]$  is identically zero).

The values of  $X_2$  and  $X_3$  converge to 5 and 8, respectively, while the upper bound established by  $\Lambda_{final}$  is 13. We observe that the values of  $X_2$  and  $X_3$  alternate, with  $X_2$  being decremented, followed by  $X_3$ , followed by  $X_2$ , and so on until convergence occurs. This demonstrates the impact of the synchronization of a node's neighbors on the synchronization of that node. Also, observe that  $X_3 > X_2$ . This is because the input node  $n_1$  is directly connected to node  $n_2$ , and hence node  $n_2$  is closer to synchronization than node  $n_3$ .

This section developed sufficient conditions for practical synchronization with a given set of input nodes. In the following section, we will propose algorithms for selecting input nodes to guarantee practical synchronization based on these conditions.

#### V. SUBMODULAR OPTIMIZATION APPROACH TO INPUT SELECTION

In this section, we present our submodular optimization approach to selecting a set of input nodes to ensure practical synchronization. Our approach is based on ensuring that the conditions for existence of a positive invariant set (Section IV-B) and convergence of the node phases to  $\Lambda_{final}$  (Section IV-C) are satisfied by establishing an equivalence between both conditions and the connectivity of an augmented network graph.

Section V-A formulates the condition for existence of a positive invariant set  $\Lambda_{PI}$  that satisfies  $\Lambda_{init} \subseteq \Lambda_{PI} \subseteq \Lambda_{bound}$  as a submodular constraint. We first define an augmented graph and prove that the conditions for existence of a positive invariant set in Theorem 3 are equivalent to the connectivity of a class of subgraphs of the augmented graph. We then show that this connectivity condition can be expressed as a constraint of the form  $h_1(A) = 0$ , where  $h_1$  is a supermodular function of  $A$ . We recall that a function  $h : 2^V \rightarrow \mathbb{R}$  is *submodular* if, for any sets  $A$  and  $B$  with  $A \subseteq B$  and any  $v \notin B$ ,

$$h(A \cup \{v\}) - h(A) \geq h(B \cup \{v\}) - h(B)$$

and that a function  $h$  is *supermodular* if  $-h$  is submodular.

Section V-B formulates convergence to the set  $\Lambda_{final}$  as a submodular constraint on the set of input nodes. As in

Section V-A, we first construct an augmented graph, and prove that the conditions identified in Theorem 4 are equivalent to connectivity of a class of subgraphs of the augmented graph (albeit a different class from that identified in Section V-A). We show that this connectivity condition is equivalent to a submodular constraint on the input set.

In Section V-C, we formulate the problem of selecting a set of input nodes to achieve practical synchronization as an optimization problem that combines the constraints of Sections V-A and V-B. We then present algorithms for selecting a set of input nodes and analyze the optimality guarantees of those algorithms.

#### A. Submodular Constraint on Input Nodes for Positive Invariance

We first define a weighted augmented network graph, denoted  $\tilde{G} = (\tilde{V}, \tilde{E})$ , and then formulate conditions for existence of a positive invariant set  $\Lambda_{PI}$  based on this graph. Let  $M > 0$  be a positive integer as in Section IV. The vertex set of  $\tilde{G}$  consists of  $M$  copies of the vertices of  $G$ , i.e.,  $\tilde{V} = \{v_m : m = 0, \dots, M, v \in V\}$ . The edge set  $\tilde{E}$  is defined by

$$\tilde{E} = \{(u_l, v_m) : (u, v) \in E, l = 0, \dots, M, m = 0, \dots, M\}.$$

The graph  $\tilde{G}$  is directed, with edge  $(u_l, v_m) \in \tilde{E}$  implying a directed edge from  $u_l$  to  $v_m$ . Each edge  $(u_l, v_m) \in \tilde{E}$  has an associated weight, defined as follows. As a preliminary, let

$$\begin{aligned} \tilde{\alpha}_{uv}(m, l) \\ \triangleq K_{uv} \left( 1 - \min \left\{ \sin \left( \frac{\theta_v^{max} m}{M} - \theta_u \right) : |\theta_u| \leq \frac{\theta_u^{max}}{M} \right\} \right). \end{aligned}$$

For each edge  $(u_l, v_m)$ , define the weight  $\tilde{\beta}_{uv}(m, l) = \tilde{\alpha}_{uv}(m, l+1) - \tilde{\alpha}_{uv}(m, l)$ . When  $l = M$ , define  $\tilde{\beta}_{uv}(m, M) = 0$ . Note that the weights  $\tilde{\beta}_{uv}(m, l)$  are nonnegative.

A class of subgraphs of  $\tilde{G}$  is defined as follows.

**Definition 8:** Let  $G'$  be a subgraph of  $\tilde{G}$ . The subgraph  $G'$  is a *class- $\mathcal{T}$*  subgraph if the neighbor set  $N'(v_m)$  of each node  $v_m$  with  $v \notin A$  satisfies

$$\sum_{u_l \in N'(v_m)} \tilde{\beta}_{uv}(m, l) > \sum_{u \in N(v)} \sum_{l=0}^{M-1} \tilde{\beta}_{uv}(m, l) - \tau_v^m, \quad (9)$$

where

$$\tau_v^m \triangleq \sum_{u \in N(v)} K_{uv} - |\omega_v| - \epsilon - \sum_{u \in N(v)} \alpha_{uv}(m, 0).$$

If no such subset exists, then  $N'(v_m) = \tilde{N}(v_m)$ . For each node  $v \in A$ , the neighbor set  $N'(v_m) = \emptyset$ .

If there exists a node  $v$  such that no neighbor set  $N'(v_m)$  satisfies (9) for all  $m = 0, \dots, M$ , then node  $v$  must be chosen as an input to ensure practical synchronization. The following theorem gives conditions for convergence based on Definition 8.

**Theorem 5:** Define  $r(v) = \lceil \frac{M\theta_v^0}{\theta_v^{max}} \rceil$ ,  $V' = \{v_{r(v)} : v \in V\}$ , and  $V'' = \{v_M : v \in V\}$ . If there exists a class- $\mathcal{T}$  subgraph in which  $v_{r(v)}$  is not connected to  $V''$  for all  $v \in V$ , then there

exists a positive invariant set  $\Lambda_{PI}$  satisfying  $\Lambda_{init} \subseteq \Lambda_{PI} \subseteq \Lambda_{bound}$ .

*Proof:* Suppose that such a class- $\mathcal{T}$  subgraph, denoted  $G'$ , exists. For each  $v \in V$ , define  $m_v^* = \max \{m : v_m \text{ connected to } v_{r(v)}\}$ , and let  $\hat{m}_v = m_v^* + 1$ . In  $G'$ ,  $N'(v_{\hat{m}_v}) \cap \{u_l : l \leq m_u^*\} = \emptyset$  (otherwise,  $v_{\hat{m}_v}$  would be connected to  $v_{r(v)}$ , a contradiction). Since the sets are disjoint and their union at most covers  $N(v_m)$ , the sum of the nonnegative weights of the edges in  $N'(v_{r(v)})$  and the edges in  $\{u_l : l \leq m_u^*\}$  is upper bounded by the sum of the edge weights in  $N(v_{r(v)})$ . Hence

$$\begin{aligned} \sum_{u_l \in N'(v_{\hat{m}_v})} \beta_{uv}(\hat{m}_v, l) + \sum_{u \in N(v)} \sum_{l=0}^{m_u^*} \tilde{\beta}_{uv}(\hat{m}_v, l) \\ \leq \sum_{u \in N(v)} \sum_{l=0}^{M-1} \tilde{\beta}_{uv}(m, l). \end{aligned}$$

Substituting (9) into the first term of the left-hand side yields

$$\begin{aligned} \sum_{u \in N(v)} \sum_{l=0}^{M-1} \tilde{\beta}_{uv}(m, l) - \tau_v^{\hat{m}_v} + \sum_{u \in N(v)} \sum_{l=0}^{m_u^*} \tilde{\beta}_{uv}(\hat{m}_v, l) \\ < \sum_{u \in N(v)} \sum_{l=0}^{M-1} \tilde{\beta}_{uv}(m, l). \end{aligned}$$

Rearranging terms and substituting the definition of  $\tau_v^{\hat{m}_v}$ , we have

$$\sum_{u \in N(v)} \sum_{l=0}^{m_u^*} \tilde{\beta}_{uv}(\hat{m}_v, l) < \sum_{u \in N(v)} K_{uv} - |\omega_v| - \sum_{u \in N(v)} \alpha_{uv}(\hat{m}_v, 0)$$

which is equivalent to

$$\begin{aligned} \sum_{u \in N(v)} \alpha_{uv}(\hat{m}_v, m_u^* + 1) &= \sum_{u \in N(v)} \alpha_{uv}(\hat{m}_v, \hat{m}_u) \\ &> \sum_{u \in N(v)} K_{uv} - |\omega_v| - \epsilon. \end{aligned}$$

Substituting the definition of  $\alpha_{uv}(\hat{m}_v, \hat{m}_u)$  and subtracting  $\sum_{u \in N(v)} K_{uv}$  from both sides yields

$$\begin{aligned} \sum_{u \in N(v)} K_{uv} \min \left\{ \sin \left( \frac{\hat{m}_v \theta_v^{max}}{M} - \theta_u \right) : |\theta_j| \leq \frac{\hat{m}_u \theta_u^{max}}{M} \right\} \\ > |\omega_v| + \epsilon. \end{aligned}$$

Hence the set

$$\Lambda_{PI} = \left\{ \theta : |\theta_v| \leq \frac{\hat{m}_v \theta_v^{max}}{M} \right\}$$

satisfies the conditions of Proposition 1, and is positive invariant. ■

We now formulate sufficient conditions for a set of input nodes to guarantee the existence of a positive invariant set, based on Theorem 5. For a class- $\mathcal{T}$  subgraph, the impact of adding a node  $v$  to the set  $A$  will be to remove the set of edges  $(u_l, v_m)$  for all  $u_l \in N'(v_m)$  and  $m = 0, \dots, M$ . Adding enough nodes to the input set will create a cut in the graph,



disconnecting  $V'$  and  $V''$ . Formally, this intuition is described by the following lemma.

**Lemma 2:** If the nodes  $\{v_m : v \in A, m = 0, \dots, M\}$  form a cut between  $V'$  and  $V''$  in any class- $\mathcal{T}$  subgraph  $G'$  of  $\tilde{G}$ , then there exists a positive-invariant set  $\Lambda_{PI}$  satisfying  $\Lambda_{init} \subseteq \Lambda_{PI} \subseteq \Lambda_{bound}$ .

*Proof:* By Definition 8, the nodes in the set  $\{v_m : m = 0, \dots, M\}$  have  $N'(v_m) = \emptyset$  in any class- $\mathcal{T}$  subgraph. Hence, if these nodes form a cut between  $V'$  and  $V''$ , then there will be no path from any node in  $V'$  to  $V''$ . By Theorem 5, there exists a positive-invariant set  $\Lambda_{PI}$  satisfying  $\Lambda_{init} \subseteq \Lambda_{PI} \subseteq \Lambda_{bound}$ . ■

The next step is to express the cut condition as a submodular constraint on the set of input nodes  $A$ . In order to construct such a constraint, define a random walk  $R[s]$  on the graph  $G'$ , in which

$$\begin{aligned} Pr(R[s] = v_m | R[s-1] = u_l) \\ = \begin{cases} \frac{1}{|\{v'_m : u_l \in N'(v'_m)\}|}, & u_l \in N'(v_m) \\ 0, & \text{else} \end{cases} \end{aligned}$$

Define  $h_1(A, G')$  to be the probability that a walk originating in  $V'$  reaches a node in  $V''$  before arriving at any node in  $\{v_m : v \in A, m = 0, \dots, M\}$ . The function  $h_1(A, G')$  can be computed efficiently using methods for computing the absorption probabilities of random walks [38]. We have the following result.

**Lemma 3:** The nodes  $\{v_m : v \in A, m = 0, \dots, M\}$  form a cut between  $V'$  and  $V''$  if and only if  $h_1(A, G') = 0$ .

*Proof:* If the nodes  $\{v_m : v \in A, m = 0, \dots, M\}$  form a cut between  $V'$  and  $V''$ , then any walk will reach  $v_m$  for some  $v \in A$  before reaching  $V''$ , and hence  $h_1(A, G') = 0$ . Conversely, if the nodes do not form a cut, then there is a sample path of the random walk that avoids all nodes in  $\{v_m : v \in A, m = 0, \dots, M\}$  before reaching  $V''$ . This implies  $h_1(A, G') > 0$ . ■

The following proposition leads to a submodular optimization formulation for ensuring positive invariance.

**Proposition 3:** The function  $h_1(A, G')$  is supermodular as a function of  $A$ .

*Proof:* To show submodularity of  $h_1(A, G')$ , consider one possible sample path of the walk. The nodes in  $\{v_m : v \in A\}$  are absorbing states of the walk. Let  $\chi(A)$  denote the indicator function of the event that the walk reaches any node in  $V''$  before an absorbing state, so that  $\chi(A) - \chi(A \cup \{v\})$  is equal to 1 if the walk reaches a node in  $V''$  before  $A$ , but reaches  $v$  before any node in  $V''$ . If  $A \subseteq B$ , then any walk that reaches  $v$  before any node in  $B$  automatically reaches  $v$  before any node in  $A$ , and hence

$$\chi(A \cup \{v\}) - \chi(A) \leq \chi(B \cup \{v\}) - \chi(B),$$

implying that the indicator function is supermodular as a function of  $A$ . Since  $h_1$  is a nonnegative weighted sum of these indicator functions,  $h_1(A, G')$  is supermodular as well. ■

Proposition 3 implies that a sufficient condition for existence of a positive invariant set is  $h_1(A, G') = 0$  for some class- $\mathcal{T}$  subgraph  $G'$ . Hence, one approach to ensuring the existence

of a positive invariant set is to generate a collection of class- $\mathcal{T}$  subgraphs and, for each subgraph, choose a set  $A$  that satisfies  $h_1(A, G') = 0$  using submodular optimization techniques. This approach will be formally developed in Section V-C.

### B. Submodular Constraint for Convergence to $\Lambda_{final}$

In this section, we develop a submodular optimization approach to selecting a set of input nodes  $A$  that satisfies the sufficient conditions for convergence to  $\Lambda_{final}$  introduced in Section IV-C. As in the previous section, our approach is to introduce an augmented graph  $\tilde{G}$ , and then provide sufficient conditions for convergence based on subgraphs of  $\tilde{G}$ .

Define the set  $\tilde{V} = \{v_m : v \in V, m = 0, \dots, M\}$ . The edge set is defined by

$$\tilde{E} = \{(u_l, v_m) : (u, v) \in E, l = 0, \dots, M, m = 0, \dots, M\}$$

The weights of the graph  $\tilde{G} = (\tilde{V}, \tilde{E})$  are defined as follows. Define

$$\begin{aligned} \alpha_{uv}(m, l) &= \min \{K_{uv} \sin(\theta_v - \theta_u) : \\ \theta_v &\in \left[ \frac{(m-1)\theta_v^{max}}{M}, \frac{m\theta_v^{max}}{M} \right], |\theta_u| \leq \frac{l\theta_u^{max}}{M} \}. \end{aligned}$$

The weight on edge  $(u_l, v_m)$  is equal to  $\beta_{uv}(m, l) = \alpha(m, l) - \alpha(m, l+1)$  for  $l = 0, \dots, (M-1)$  and  $\beta_{uv}(m, M) = 0$ . By definition of  $\alpha_{uv}(m, l)$ , the weights  $\beta_{uv}(m, l)$  are nonnegative.

We now define a class of subgraphs of  $\tilde{G} = (\tilde{V}, \tilde{E})$  that will be used to derive sufficient conditions for convergence.

**Definition 9:** A subgraph  $G'' = (\tilde{V}, E'')$  is a class- $\mathcal{U}$  subgraph if the neighbor set of each node  $v_m$ , denoted  $N''(v_m)$ , satisfies

$$\sum_{u_l \in N''(v_m)} \beta_{uv}(m, l) \geq \tau_v^m, \quad (10)$$

where

$$\tau_v^m = \sum_{u \in N(v)} \sum_{l=1}^M \beta_{uv}(m, l) - \left( |\omega_v| - \sum_{u \in N(v)} \alpha_{uv}(m, M) \right).$$

The following result gives a sufficient condition for convergence of  $\theta(t)$  to  $\Lambda_{final}$  based on the connectivity of class- $\mathcal{U}$  subgraphs of  $\tilde{G}$ .

**Theorem 6:** Suppose that a set  $\Lambda_{PI} = \{\theta : |\theta_v| \leq \theta_v^{PI}\}$  exists that is positive invariant and satisfies  $\Lambda_{init} \subseteq \Lambda_{PI} \subseteq \Lambda_{bound}$ . Let  $r(v) = \lceil \frac{M\gamma}{\theta_v^{max}} \rceil$  and  $s(v) = \lceil \frac{M\theta_v^{PI}}{\theta_v^{max}} \rceil$ . Define  $V' = \{v_{r(v)} : v \in V\}$  and  $V'' = \{v_{s(v)} : v \in V\}$ . If, for each node  $v_m \in V'$ , there is a node  $u_l \in V''$  and a directed path from  $u_l$  to  $v_m$ , then for any  $\theta(0) \in \Lambda_{init}$ ,  $\theta(t) \in \Lambda_{final}$  for  $t$  sufficiently large.

*Proof:* The approach of the proof will be to assume that  $\theta(t)$  does not converge to  $\Lambda_{final}$ , and then use Algorithm 2 to construct a class- $\mathcal{U}$  subgraph in which  $V''$  is not connected to  $V'$ . Define  $\mathbf{X}^*$  to be the vector  $\mathbf{X}[k]$  from Algorithm 2 when the algorithm converges.

A subgraph  $G''$  is constructed as follows. For each node  $v_m$  with  $m < X_v^*$ , define  $N''(v_m) = \{u_l : u \in N(v), l = 0, \dots, X_u^* - 1\}$ . For each node  $v_m$  with  $m \geq X_v^*$ , select an arbitrary set  $N''(v_m)$  that satisfies (10).

We now show that  $G''$  is a class- $\mathcal{U}$  subgraph. By construction, the sets  $N''(v_m)$  for  $m \geq X_v^*$  satisfy (10), so it remains to prove that Eq. (10) holds for  $\{v_m : m < X_v^*\}$ . For all such  $v_m$ ,

$$\sum_{u_l \in N''(v_m)} \beta_{uv}(m, l) = \sum_{u \in N(v)} \sum_{l=0}^{X_u^*-1} \beta_{uv}(m, l).$$

Now  $N(v_m) = N''(v_m) \cup \{(u_l, v_m) : l \geq X_u^*\}$  and this is a disjoint union. Hence

$$\begin{aligned} & \sum_{u_l \in N''(v_m)} \beta_{uv}(m, l) \\ &= \sum_{u_l \in N(v_m)} \beta_{uv}(m, l) - \sum_{u \in N(v)} \sum_{l=X_u^*}^M \beta_{uv}(m, l) \\ &= \sum_{u_l \in N(v_m)} \beta_{uv}(m, l) \\ &\quad - \sum_{u \in N(v)} \sum_{l=X_u^*}^{M-1} (\alpha_{uv}(m, l) - \alpha_{uv}(m, l+1)) \\ &= \sum_{u_l \in N(v_m)} \beta_{uv}(m, l) \\ &\quad - \sum_{u \in N(v)} (\alpha_{uv}(m, X_u^*) - \alpha_{uv}(m, M)) \end{aligned} \quad (11)$$

Since Algorithm 2 terminates with  $X_v^* > m$ , we must have

$$\sum_{u \in N(v)} \alpha_{uv}(m, X_u^*) < |\omega_v| + \epsilon.$$

Substituting this inequality into (11) yields

$$\begin{aligned} & \sum_{u_l \in N''(v_m)} \beta_{uv}(m, l) \\ &> \sum_{u_l \in N(v_m)} \beta_{uv}(m, l) - \left( |\omega_v| + \epsilon - \sum_{u \in N(v)} \alpha_{uv}(m, M) \right). \end{aligned}$$

Hence  $N''(v_m)$  satisfies (10), and  $G''$  is a class- $\mathcal{U}$  subgraph.

By construction, there is no path from any node in  $\{v_m : m \geq X_v^*\}$  to any node in  $\{v_m : m < X_v^*\}$ . Since convergence to  $\Lambda_{final}$  is not guaranteed, by Theorem 4, there exists  $v$  with  $r(v) < X_v^*$ , and hence  $V' \cap \{v_m : m < X_v^*\} \neq \emptyset$ . At the same time,  $V'' \subseteq \{v_m : m \geq X_v^*\}$ , since  $X_v[0] = s(v)$  and  $X_v$  is nonincreasing at each iteration of Algorithm 2. Hence there exists at least one  $v$  with  $v_{r(v)}$  not path-connected to any node in  $V''$ , completing the proof. ■

The following proposition established a converse result to Theorem 6. We show that, if the sufficient conditions defined for Algorithm 2 in Theorem 4 hold, then  $V''$  is connected to  $V'$  in any class- $\mathcal{U}$  subgraph.

**Proposition 4:** If  $X_v[k] = m$  in Algorithm 2 at some iteration  $k \geq 0$ , then there exists a path from at least one node in  $V''$  to  $v_m$  in every class- $\mathcal{U}$  subgraph of  $\tilde{G}$ .

*Proof:* The proof is by induction on  $k$ . By definition of the algorithm,  $X_v[0] = s(v)$ , and  $v_{s(v)} \in V''$ , implying that connectivity trivially holds when  $k = 0$ .

Suppose that, at iteration  $k$ , node  $v$  satisfies  $X_v[k] = X_v[k-1] - 1$ . Let  $m = X_v[k]$  and define  $G''$  to be a class- $\mathcal{U}$  subgraph, in which the neighbor set of  $v_m$  is  $N''(v_m)$ . By definition of Algorithm 2,

$$\begin{aligned} |\omega_v| + \epsilon &< \sum_{u \in N(v)} \alpha_{uv}(m, X_u[k-1]) \\ &= \sum_{u \in N(v)} \left[ \sum_{l=X_u[k-1]}^{M-1} (\alpha_{uv}(m, l) - \alpha_{uv}(m, l+1)) \right. \\ &\quad \left. + \alpha_{uv}(m, M) \right] \end{aligned}$$

and rearranging terms yields

$$\sum_{u \in N(v)} \sum_{l=X_u[k-1]}^M \beta_{uv}(m, l) > |\omega_v| + \epsilon - \sum_{u \in N(v)} \alpha_{uv}(m, M).$$

We then have

$$\begin{aligned} & \sum_{u_l \in N''(v_m)} \beta_{uv}(m, l) + \sum_{u \in N(v)} \sum_{l=X_u[k-1]}^M \beta_{uv}(m, l) \\ &> \sum_{u_l \in N(v_m)} \beta_{uv}(m, l) - \left( |\omega_v| + \epsilon - \sum_{u \in N(v)} \alpha_{uv}(m, M) \right) \\ &\quad + |\omega_v| + \epsilon - \sum_{u \in N(v)} \alpha_{uv}(m, M) \\ &= \sum_{u_l \in N(v_m)} \beta_{uv}(m, l). \end{aligned}$$

Since

$$N''(v_m) \cup \{u_l : l \geq X_u[k-1]\} \subseteq N(v_m)$$

and the weights  $\beta_{uv}(m, l)$  are nonnegative, there exists

$$u_l \in N''(v_m) \cap \{u_l : l \geq X_u[k-1]\}.$$

By induction, each node in  $\{u_l : l \geq X_u[k-1]\}$  is connected to  $V''$  in  $G''$ , and hence  $v_m$  is connected to  $V''$ . ■

Proposition 4 leads to the following bound on the function  $g(A)$  introduced in Section III-B.

**Corollary 1:** The function  $g(A)$  satisfies

$$g(A) \leq \max_v \left\{ \min_m \left\{ \frac{m\theta_v^{max}}{M} : v_m \text{ connected to } V'' \right\} \right\} \quad \text{in all class-}\mathcal{U} \text{ subgraphs} \}. \quad (12)$$

*Proof:* By Proposition 4, if  $v_m$  is connected to  $V''$  in all class- $\mathcal{U}$  subgraphs, then  $|\theta_v^*| \leq \frac{m\theta_v^{max}}{M}$  for any equilibrium point  $\theta^*$ . Eq. (12) follows from the fact that  $g(A)$  is defined as the largest value of  $\gamma$  such that  $|\theta_v^*| = \gamma$  for some equilibrium  $\theta^*$  and node  $v$ . ■

We now show that convergence to  $\Lambda_{final}$  can be mapped to a submodular constraint on the set of input nodes  $A$ . Let  $\pi$  be a probability distribution on the set of class- $\mathcal{U}$  subgraphs with  $\pi(G') > 0$  for each class- $\mathcal{U}$  subgraph  $G'$ . Define the function  $h_2(A)$  as

$$h_2(A) \triangleq \sum_{G' \in \mathcal{U}} \pi(G') f(A|G'),$$

where  $f(A|G')$  is defined by

$$f(A|G') = |\{v_m \in V' : \text{there exists } u_l \in V'' \text{ and a path from } u_l \text{ to } v_m\}|.$$

The function  $h_2(A)$  can be approximated by randomly sampling a collection of class- $\mathcal{U}$  subgraphs and computing  $f(A|G')$  for each subgraph. The following corollary describes convergence to  $\Lambda_{final}$  based on  $h_2(A)$ .

*Corollary 2:* If  $h_2(A) = n$ , then for any  $\theta(0) \in \Lambda_{init}$ ,  $\theta(t) \in \Lambda_{final}$  for  $t$  sufficiently large.

*Proof:* We have that  $f(A|G') \leq n$  and the function  $h_2(A)$  is a convex combination of the values of  $f(A|G')$ . Hence  $h_2(A) \leq n$  with equality if and only if  $f(A|G') = n$  for all class- $\mathcal{U}$  subgraphs  $G'$ . By definition of  $f(A|G')$ , for every node  $v_m \in V'$ , there exists a path from  $V''$  to  $v_m$  in each class- $\mathcal{U}$  subgraph. Convergence to  $\Lambda_{final}$  is then implied by Theorem 6. ■

Finally, to show that  $h_2(A) = n$  is a submodular constraint in  $A$ , we have the following result establishing submodularity of  $h_2(A)$ .

*Proposition 5:* The function  $h_2(A)$  is submodular as a function of  $A$ .

*Proof:* We first show that  $f(A|G')$  is submodular as a function of  $A$ . The impact of adding a node  $\{v\}$  to  $A$  is to increase the set  $V''$  to  $V'' \cup \{v_m : m \geq 0\}$ . Hence, the increment  $f(A \cup \{v\}) - f(A)$  is equal to the number of nodes in  $V'$  that are connected to  $\{v_m : m \geq 0\}$  but are not connected to  $V''$ . If  $A \subseteq B$ , then any node that is not connected to  $V''$  with input set  $B$  is automatically not connected to  $V''$  with input set  $A$ . Hence

$$f(A \cup \{v\}|G') - f(A|G') \geq f(B \cup \{v\}|G') - f(B|G'),$$

establishing submodularity of  $f(A|G')$ . The function  $h_2(A)$  is equal to a nonnegative weighted sum of submodular functions, and hence is submodular. ■

Proposition 5 implies that convergence to  $\Lambda_{final}$  can be formulated as a submodular constraint  $h_2(A) = n$ , which ensures that all nodes in  $V'$  are connected to  $V''$ . In the following section, we demonstrate how to combine the submodular constraints of Sections V-A and V-B to formulate selection of input nodes for practical synchronization as a submodular optimization problem.

### C. Submodular Optimization Approach to Selecting Inputs

The conditions derived in Sections V-A and V-B imply that the nodes achieve practical synchronization if  $h_1(A|G) = 0$  for some class- $\mathcal{T}$  subgraph  $G$  and  $h_2(A) = n$ . We consider two input selection problems. In the first problem, we select the minimum-size set of input nodes in order to guarantee  $\gamma$ -practical synchronization. In the second problem, we select a set of up to  $k$  input nodes in order to maximize the input cohesiveness  $g(A)$ .

The problem of selecting the minimum-size input nodes in order to guarantee  $\gamma$ -practical synchronization can be formu-

lated as

$$\begin{aligned} & \text{minimize} && |S| \\ & \text{s.t.} && h_1(A|G') = 0 \text{ for some class-}\mathcal{T} \text{ subgraph } G' \\ & && h_2(A) = n \end{aligned} \quad (13)$$

In solving (13), we first observe that the function  $h_2(A)$  depends on the positive invariant set  $\Lambda_{PI}$ . Hence, we propose a sequential approach, in which a set of positive invariant sets, and the minimum-size set of input nodes required to ensure positive invariance, are computed. A minimum-size set of additional input nodes required to ensure convergence to  $\Lambda_{final}$  is then computed.

The first step in the procedure is to generate a random collection of class- $\mathcal{T}$  subgraphs  $\{G'_1, \dots, G'_N\}$ . For each subgraph, the optimization problem

$$\text{minimize}\{|A| : h_1(A|G_i) = 0\} \quad (14)$$

is approximated. One for solving (14) is as follows. Initialize a set  $A_i = \emptyset$ . At each iteration, select a node  $\{v\}$  such that  $h(A_i) - h(A_i \cup \{v\})$  is maximized, terminating when  $h(A_i) = 0$ .

After obtaining sets  $A_1, \dots, A_N$ , the next step in the procedure is to ensure that the condition  $h_2(A) = n$  is satisfied. This condition can be ensured by solving

$$\text{minimize}\{|A| : h_2(A \cup A_i) = n\} \quad (15)$$

for  $i = 1, \dots, N$ , resulting in a collection  $A'_1, \dots, A'_N$ . The set  $A'_j$  with minimum cardinality is then chosen as the minimum-size set for guaranteeing practical synchronization. This procedure is shown as Algorithm 3.

*Lemma 4:* Algorithm 3 returns a set  $A$  that guarantees practical synchronization.

*Proof:* The set  $A$  returned by Algorithm 3 can be decomposed as  $A = A_i \cup A'_i$  for some index  $i$ . The set  $A_i$  satisfies  $h_1(A_i|G_i) = 0$  for the class- $\mathcal{T}$  subgraph  $G_i$ , and hence there exists a positive invariant subset  $\Lambda_{PI}$  satisfying  $\Lambda_{init} \subseteq \Lambda_{PI} \subseteq \Lambda_{bound}$  when the input set is  $A$ . Furthermore,  $h_2(A_i \cup A'_i) = n$ , and hence convergence to  $\Lambda_{final}$  is guaranteed by Theorem 6. Combining these two conditions leads to guaranteed practical synchronization by Theorem 2. ■

We next analyze the optimality guarantees provided by this approach.

*Lemma 5:* For each  $i = 1, \dots, N$ , the sets  $A_i$  and  $A'_i$  selected by Algorithm 3 satisfy

$$\frac{|A_i|}{|A_i^*|} \leq 1 + \ln n \quad (16)$$

$$\frac{|A'_i|}{|A_i'^*|} \leq 1 + \ln \left\{ \frac{n}{h_2(A_i \cup A'_i) - h_2(A_i \cup \hat{A}'_i)} \right\} \quad (17)$$

where  $A_i^*$  and  $A_i'^*$  denote the optimal solutions to (14) and (15), respectively and  $\hat{A}'_i$  is the value of  $A'_i$  at the second-to-last iteration of the second inner loop in Algorithm 3.

*Proof:* Problems (14) and (15) involve minimizing the cardinality of a set subject to a submodular constraint. For the problem of minimizing  $|A|$  subject to a constraint that

**Algorithm 3** Algorithm for selecting the minimum-size input set  $A$  to guarantee practical synchronization.

---

```

1: procedure MIN_INPUT_PS( $G = (V, E)$ ,  $\omega_1, \dots, \omega_n$ ,  $\gamma$ ,
    $\{\theta_1^{max}, \dots, \theta_n^{max}\}$ ,  $\{\theta_1^0, \dots, \theta_n^0\}$ )
2:   Input: Graph  $G = (V, E)$ , intrinsic frequencies
    $\omega_1, \dots, \omega_n$ , bound  $\gamma$ , boundary values  $\{\theta_i^{max} : i \in V\}$ ,
   initial states  $\{\theta_i^0 : i \in V\}$ .
3:   Output: Set of input nodes  $A$ 
4:   Generate a collection of class- $\mathcal{T}$  subgraphs
    $G_1, \dots, G_N$ 
5:   for  $i = 1, \dots, N$  do
6:      $A_i \leftarrow \emptyset$ 
7:     while  $h_1(A_i | G'_i) > 0$  do
8:        $v \leftarrow \arg \min \{h_1(A_i \cup \{v\} | G'_i) : v \in V\}$ 
9:        $A'_i \leftarrow \emptyset$ 
10:    end while
11:     $(\bar{\theta}_1^{(i)}, \dots, \bar{\theta}_n^{(i)}) \leftarrow \text{Identify\_PI\_Set}(G,$ 
    $(\omega_1, \dots, \omega_n), A'_i, \{\theta_v^0 : v \in V\}, \{\theta_v^{max} : v \in V\})$ 
12:     $A'_i \leftarrow \emptyset$ 
13:    while  $h_2(A_i \cup A'_i) < n$  do
14:       $v \leftarrow \arg \max \{h_2(A_i \cup A'_i \cup \{v\}) : v \in V\}$ 
15:       $A'_i \leftarrow A'_i \cup \{v\}$ 
16:    end while
17:  end for
18:   $j \leftarrow \arg \min \{|A_i \cup A'_i| : i = 1, \dots, N\}$ 
19:   $A \leftarrow A_j \cup A'_j$ ; return  $A$ 
20: end procedure

```

---

$\{f(A) \leq \alpha\}$  for some  $\alpha$ , the greedy algorithm is known to return a set  $A$  satisfying

$$\frac{|A|}{|A^*|} \leq 1 + \ln \left\{ \frac{\alpha}{f(A) - f(\hat{A})} \right\},$$

where  $\hat{A}$  is the set computed by the greedy algorithm at the second-to-last iteration. This bound reduces to (16) and (17) for the problems (14) and (15), respectively. ■

The runtime of Algorithm 3 is determined by the network size  $n$  and the number of class- $\mathcal{T}$  subgraphs generated, and is bounded by  $O(n^2 N)$  evaluations of  $h_1(A | G')$  and  $h_2(A)$ .

We now consider the problem of selecting up to  $k$  input nodes in order to maximize the input cohesiveness  $g(A)$ , formulated as

$$\begin{aligned}
& \text{maximize } g(A) \\
& \text{s.t. } |A| \leq k \\
& \quad h_1(A | G') = 0 \text{ for some class-}\mathcal{T} \text{ subgraph } G'
\end{aligned} \tag{18}$$

In order to approximate the solution to (18), we define the function  $h_3(A)$  to be the expected number of nodes connected to  $V''$  in a random class- $\mathcal{U}$  subgraph, i.e.,

$$h_3(A) = \mathbf{E}(|\{v_m : \text{there exists a node } u_l \in V'' \text{ connected to } v_m\}|).$$

The definition of  $h_3(A)$  is motivated by Corollary 1, which establishes an upper bound on  $g(A)$  based on the number of nodes that are connected to  $V''$ .

By the same argument as Proposition 5,  $h_3(A)$  is submodular as a function of  $A$ . We consider the problem of maximizing  $h_3(A)$ , subject to the constraint that  $h_2(A | G') = 0$  for some class- $\mathcal{T}$  subgraph  $G'$  and  $|A| \leq k$ . Our approach is a sequential algorithm in which a collection of subsets  $A_1, \dots, A_N$  is found that guarantee the existence of a positive invariant set in a collection of class- $\mathcal{T}$  subgraphs  $G'_1, \dots, G'_N$ . For each subset of  $A_i$ , we then select a subset  $A'_i$  such that  $|A_i \cup A'_i| \leq k$  and  $h_3(A_i \cup A'_i)$  is maximized. A formal description is given as Algorithm 4.

**Algorithm 4** Algorithm for selecting the minimum-size input set  $A$  to guarantee practical synchronization.

---

```

1: procedure MAXIMIZE_SYNC( $G = (V, E)$ ,  $\omega_1, \dots, \omega_n$ ,
    $k$ ,  $\{\theta_1^{max}, \dots, \theta_n^{max}\}$ ,  $\{\theta_1^0, \dots, \theta_n^0\}$ )
2:   Input: Graph  $G = (V, E)$ , intrinsic frequencies
    $\omega_1, \dots, \omega_n$ , number of input nodes  $k$ , boundary values
    $\{\theta_i^{max} : i \in V\}$ , initial states  $\{\theta_i^0 : i \in V\}$ .
3:   Output: Set of input nodes  $A$ 
4:   Generate a collection of class- $\mathcal{T}$  subgraphs
    $G_1, \dots, G_N$ 
5:   for  $i = 1, \dots, N$  do
6:      $A_i \leftarrow \emptyset$ 
7:     while  $h_1(A_i | G'_i) > 0$  do
8:        $v \leftarrow \arg \min \{h_1(A_i \cup \{v\} | G'_i) : v \in V\}$ 
9:        $A'_i \leftarrow \emptyset$ 
10:    end while
11:     $(\bar{\theta}_1^{(i)}, \dots, \bar{\theta}_n^{(i)}) \leftarrow \text{Identify\_PI\_Set}(G,$ 
    $(\omega_1, \dots, \omega_n), A'_i, \{\theta_v^0 : v \in V\}, \{\theta_v^{max} : v \in V\})$ 
12:     $A'_i \leftarrow \emptyset$ 
13:    if  $|A_i| < k$  then
14:      while  $|A_i \cup A'_i| \leq k$  do
15:         $v \leftarrow \arg \max \{h_3(A_i \cup A'_i \cup \{v\}) : v \in V\}$ 
16:         $A'_i \leftarrow A'_i \cup \{v\}$ 
17:      end while
18:    end if
19:  end for
20:   $j \leftarrow \arg \max \{h_3(A_i \cup A'_i) : |A_i| \leq k\}$ 
21:   $A \leftarrow A_j \cup A'_j$ ; return  $A$ 
22: end procedure

```

---

The following lemma describes the optimality guarantees of this approach.

*Lemma 6:* Define

$$A_i^* = \arg \max_{A'_i} \{h_3(A_i \cup A'_i) : |A_i \cup A'_i| \leq k\},$$

and let  $A$  denote the set returned by Algorithm 4. Then

$$h_3(A) \geq (1 - 1/e) h_3(A \cup A_i^*). \tag{19}$$

*Proof:* For the problem of selecting a set  $A$  of cardinality  $k$  to maximize a monotone submodular objective function, the greedy algorithm is known to provide an optimality bound of  $(1 - 1/e)$ . The problem of maximizing  $h_3(A)$  subject to a constraint on  $k$  has this structure, leading to the optimality bound (19). ■

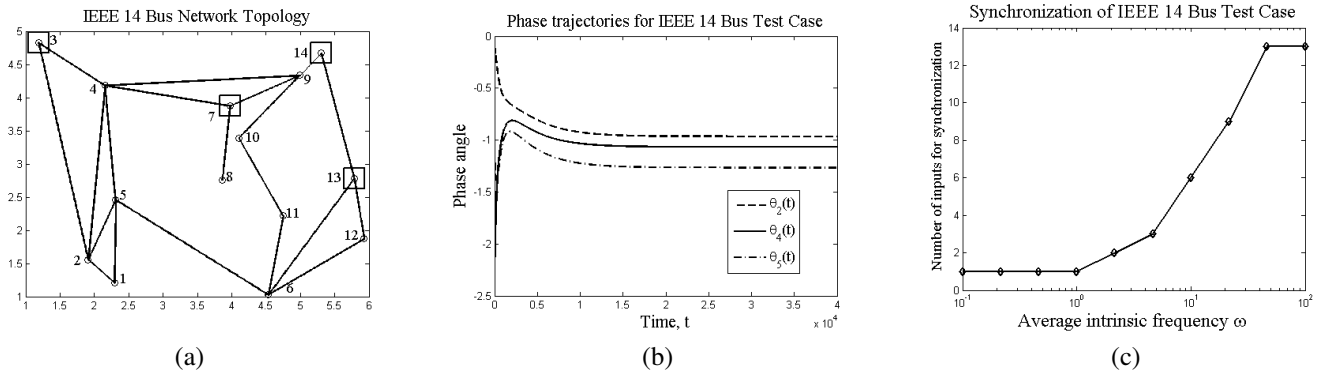


Fig. 3. Synchronization case study using the IEEE 14 Bus case study [15]. Coupling coefficients were chosen to be equal to the inverse of the magnitudes of the line impedances, while the intrinsic frequencies were chosen from a Gaussian distribution. (a) Illustration of 14-bus network and inputs needed for one trial when the variance of the intrinsic frequencies was equal to 8. The black squares indicate input nodes. (b) Sample trajectories for nodes 2, 4, and 5 in the 14-bus network. Node trajectories converge to within the desired phase difference ( $\gamma = \pi/3$ ) of each other. (c) Number of inputs required for practical edge synchronization as a function of the variance of the intrinsic frequencies. As the variance increases, the frequencies of neighboring nodes diverge and hence more inputs are needed to ensure synchronization.

In this case, the runtime is  $O(knN)$  evaluations of  $h_1(A|G')$  and  $h_3(A)$ . Hence the problem of selecting a minimum-size set of input nodes to guarantee practical synchronization (Eq. (13)), as well as the problem of selecting a fixed-size set of input nodes to maximize input cohesiveness (Eq. (18)), can be approximated within the submodular optimization framework.

## VI. NUMERICAL RESULTS

We investigated our approach through a numerical study. The goal of our study was to analyze the impact of the external inputs on the oscillator dynamics, as well as to investigate the properties of the inputs chosen by our algorithm. We considered synchronization in three application domains: (i) synchronization of phase angles in the IEEE 14 Bus power system test case [15], (ii) time synchronization in wireless networks [35], modeled as a geometric random graph, and (iii) synchronization of firing rates in the *C. Elegans* neuronal network [16]. The results of each study are summarized as follows. In all cases, the parameters  $M = N = 20$ .

### A. Phase Synchronization in Power System

Synchronization plays a vital role in the power system, in which stable operation requires buses and generators to maintain the same frequency and a relative phase difference of no more than  $\pi/2$  on each transmission line (edge) [39]. In order to evaluate our approach to synchronization of power systems, we consider the first-order model (2) with negligible resistance on the transmission lines. The input nodes represent generators that are fixed to a common reference phase and frequency in order to restore stability to the grid. We studied phase synchronization in the power system using the IEEE 14 Bus case study [15], which provides a network topology with  $n = 14$  nodes along with the line impedances of the edges. We defined the coupling coefficient  $K_{uv} = 1/\eta_{uv}$ , where  $\eta_{uv}$  is the magnitude of the impedance of edge  $(u, v)$ . The intrinsic frequencies were chosen according to a zero-mean Gaussian distribution with variance ranging from 0.01 to 100 in different trials. The goal was to ensure that the phase angles

satisfied  $|\theta_v - \theta_u| < \frac{\pi}{3}$  (i.e.,  $\gamma$ -practical edge synchronization with  $\gamma = \pi/3$ ), while ensuring that  $|\theta_v(t)| \leq \pi/4$  for all  $t$ , consistent with the goal of guaranteeing power system stability [39].

Figure 3(a) illustrates input selection for a given set of intrinsic frequencies. The variance of the intrinsic frequencies was equal to 8. The dark squares indicate input nodes selected by our algorithm ( $|A| = 4$ ). We observe that each non-input node is at most two hops away from an input node, which suggests that centrally located nodes are more likely to be candidates for inputs. On the other hand, nodes with high degree were not necessarily chosen as inputs. Additional information beyond the network topology, including the intrinsic frequencies and the coupling coefficients between nodes, is incorporated into the input selection process.

In Figure 3(b), trajectories of non-input nodes 2, 4, and 5 from Figure 3(a) are shown. The trajectories of these neighboring nodes converge to within the desired bound of  $\frac{\pi}{3}$  from each other, and synchronize to the frequency of the input nodes ( $\omega_0 = 0$ ; by Lemma 1, an arbitrary initial frequency could be chosen, but we selected  $\omega_0 = 0$  for clarity of the figure).

Figure 3(c) shows the number of input nodes required to achieve practical synchronization for different intrinsic frequencies. When the variance of  $\{\omega_v : v \in V\}$  is low, only one input node is required to achieve synchronization. For variance near 10, an intermediate number of inputs is required for synchronization, while for large variance in the intrinsic frequencies all nodes must be selected as input nodes to guarantee synchronization.

### B. Time Synchronization in Wireless Network

We analyzed time synchronization in wireless networks, in which the input nodes represent anchor nodes with clocks that are set to the reference time. We modeled the wireless network as a geometric random graph. Each node was assumed to share an edge with other nodes within a range of 100m. The simulation was run with the number of nodes  $n = \{20, 50, 100\}$ . The nodes were assumed to be deployed uniformly at random

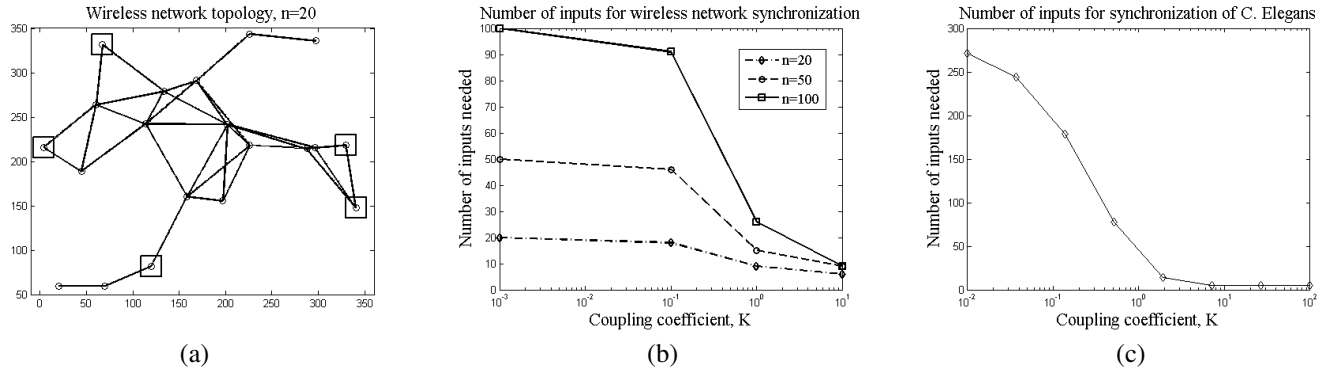


Fig. 4. Numerical results for time synchronization and neuronal synchronization. (a) Illustration of inputs chosen for a wireless network, modeled as a geometric random graph, when  $K = 0.7197$ ,  $n = 20$ , and the input nodes are indicated by squares. The set of input nodes is based on the network topology and intrinsic frequencies of the nodes. (b) Number of input nodes required for time synchronization. The number of nodes is a decreasing function of the coupling coefficient. The reduction in the fraction of input nodes required for synchronization is most significant for larger network size. (c) Number of input nodes required for synchronization of the *C. Elegans* network ( $n = 279$ ) when the intrinsic frequencies have Gaussian distribution with unit variance. The number of inputs is a decreasing function of the coupling coefficient.

over a square region, with area chosen so that each node had three neighbors on average. Nodes were assumed to have i.i.d. Gaussian intrinsic frequency with unit variance and zero mean. We computed the number of input nodes required to ensure  $\gamma$ -practical node synchronization with  $\gamma = \pi/5$  of each other. The edges had identical coupling with  $K_{uv} = K$  for all  $(u, v) \in E$ , with  $K$  in the set  $\{0.01, 0.01, 0.1, 1, 10\}$ .

Figure 4(a) shows one example of a geometric random graph with  $n = 20$  and  $K = 0.7197$  for all links. The input nodes are indicated by black squares. As in the power system case study, most non-input nodes are within two hops of an input node, and high-degree nodes are not necessarily chosen as inputs. Note that two adjacent nodes are both chosen as inputs, due to the high intrinsic frequencies of those nodes.

The number of nodes required for synchronization, shown in Figure 4(b), is a decreasing function of the coupling coefficient, since a stronger coupling implies that input nodes are able to steer their neighbors towards synchronization. We observe that the fraction of input nodes required for synchronization is smaller for larger networks.

### C. Firing Rate Synchronization in Neuronal Network

Synchronization of groups of neuronal cells to a common frequency plays a role in information storage and processing for motor control [40] and memory [3]. We analyzed the impact of input nodes on synchronization using the *Caenorhabditis Elegans* (roundworm) neuronal network [16]. The number of nodes is  $n = 279$ . The intrinsic frequencies were independently from a zero-mean, unit-variance Gaussian distribution. All links had the same coupling coefficient  $K$ , which varied from  $K = 0.001$  to  $K = 10$ . The goal was to achieve practical node synchronization with  $\gamma = \pi/4$ .

We observed that the number of input nodes required decreases as the coupling coefficient increases. The reduction in the number of nodes required was larger than in the random geometric graph (Section VI-B), even though the neuronal network contained more nodes. An explanation for this phenomenon is that the neuronal network has higher degree, as well as additional links between different regions

of the network. This more connected network topology enables synchronization of more nodes and at a lower coupling coefficient.

## VII. CONCLUSIONS AND FUTURE WORK

We considered the problem of ensuring practical synchronization of phase-coupled oscillators from any initial state by introducing external input nodes. We studied two synchronization problems. In the first problem, the goal is for all node phases to converge to within a bound of a given reference phase, while also converging to the same frequency. In the second problem, the relative phase differences between neighboring nodes must converge to a given bound, while all nodes converge to the same frequency.

We derived sufficient conditions for achieving both synchronization goals. We proved that practical synchronization is satisfied if two conditions hold. First, if there exists a positive invariant set  $\Lambda_{PI}$  that contains the set of initial node states and satisfies  $\Lambda_{PI} \subseteq \Lambda_{bound}$ , where  $\Lambda_{bound}$  defines the set of upper bounds on the node phases. Second, for any initial state inside  $\Lambda_{PI}$ , each node's phase is guaranteed to converge to within the desired level of synchronization. We developed efficient algorithms for verifying both conditions.

We formulated a submodular optimization approach for selecting a set of input nodes to guarantee synchronization. To achieve a submodular formulation, we first constructed an augmented network graph. We identified two classes of subgraphs (denoted class- $\mathcal{T}$  and class- $\mathcal{U}$ ) of the augmented graph. We proved that the sufficient condition for existence of a positive invariant set is equivalent to the existence of a cut in at least one class- $\mathcal{T}$  subgraph. We then showed that the sufficient condition for convergence to practical synchronization is equivalent to a connectivity constraint for over all class- $\mathcal{U}$  subgraphs.

We mapped this condition to submodular constraints on the set of input nodes. Based on the submodular formulation, we proposed efficient algorithms with provable optimality bounds for selecting a set of up to  $k$  nodes in order to maximize the level of synchronization in the network, as well as selecting

the minimum-size input set to guarantee a desired level of synchronization.

Our approach was validated through a numerical study of synchronization in power system, wireless, and neuronal networks. Our numerical study supports the intuition that the number of input nodes required decreases as the coupling between neighboring nodes grows stronger. In addition, we found that centrally located nodes are often chosen as inputs, so that the maximum distance between a node and the input set is small.

The threshold-based conditions derived in this work are sufficient, but not necessary. Characterizing the space of networks where these conditions are also necessary, as well as developing tighter sufficient conditions, remains an open problem. Furthermore, while the model we studied considers the first-order dynamics of the nodes, second-order dynamics are often used to model systems such as the power grid [24]. Generalizing our approach to these second-order systems is a direction of future research.

## REFERENCES

- [1] F. Dörfler and F. Bullo, "Synchronization and transient stability in power networks and nonuniform Kuramoto oscillators," *SIAM Journal on Control and Optimization*, vol. 50, no. 3, pp. 1616–1642, 2012.
- [2] R. Eckhorn, R. Bauer, W. Jordan, M. Brosch, W. Kruse, M. Munk, and H. Reitboeck, "Coherent oscillations: A mechanism of feature linking in the visual cortex?" *Biological Cybernetics*, vol. 60, no. 2, pp. 121–130, 1988.
- [3] W. Klimesch, "Memory processes, brain oscillations and EEG synchronization," *International Journal of Psychophysiology*, vol. 24, no. 1, pp. 61–100, 1996.
- [4] D. A. Paley, N. E. Leonard, R. Sepulchre, D. Grunbaum, and J. K. Parrish, "Oscillator models and collective motion," *IEEE Control Systems*, vol. 27, no. 4, pp. 89–105, 2007.
- [5] A. Arenas, A. Díaz-Guilera, J. Kurths, Y. Moreno, and C. Zhou, "Synchronization in complex networks," *Physics Reports*, vol. 469, no. 3, pp. 93–153, 2008.
- [6] M. G. Rosenblum, A. S. Pikovsky, and J. Kurths, "Phase synchronization of chaotic oscillators," *Physical Review Letters*, vol. 76, no. 11, p. 1804, 1996.
- [7] J. A. Acebrón, L. L. Bonilla, C. J. P. Vicente, F. Ritort, and R. Spigler, "The Kuramoto model: A simple paradigm for synchronization phenomena," *Reviews of Modern Physics*, vol. 77, no. 1, p. 137, 2005.
- [8] S. H. Strogatz, "From Kuramoto to Crawford: exploring the onset of synchronization in populations of coupled oscillators," *Physica D: Nonlinear Phenomena*, vol. 143, no. 1, pp. 1–20, 2000.
- [9] A. Jadbabaie, N. Motee, and M. Barahona, "On the stability of the Kuramoto model of coupled nonlinear oscillators," *Proceedings of the American Control Conference (ACC)*, vol. 5, pp. 4296–4301, 2004.
- [10] N. Chopra and M. W. Spong, "On exponential synchronization of Kuramoto oscillators," *IEEE Transactions on Automatic Control*, vol. 54, no. 2, pp. 353–357, 2009.
- [11] L. M. Childs and S. H. Strogatz, "Stability diagram for the forced Kuramoto model," *Chaos: An Interdisciplinary Journal of Nonlinear Science*, vol. 18, 2008.
- [12] C. C. McIntyre and P. J. Hahn, "Network perspectives on the mechanisms of deep brain stimulation," *Neurobiology of Disease*, vol. 38, no. 3, pp. 329–337, 2010.
- [13] Y. Wang and F. J. Doyle, "Exponential synchronization rate of Kuramoto oscillators in the presence of a pacemaker," *IEEE Transactions on Automatic Control*, vol. 58, no. 4, pp. 989–994, 2013.
- [14] H. Kori and A. S. Mikhailov, "Entrainment of randomly coupled oscillator networks by a pacemaker," *Physical Review Letters*, vol. 93, no. 25, p. 254101, 2004.
- [15] "Data sheets for IEEE 14 bus system," [http://shodhganga.inflibnet.ac.in/bitstream/10603/5247/18/19\\_appendix.pdf](http://shodhganga.inflibnet.ac.in/bitstream/10603/5247/18/19_appendix.pdf).
- [16] L. R. Varshney, B. L. Chen, E. Paniagua, D. H. Hall, and D. B. Chklovskii, "Structural properties of the *Caenorhabditis Elegans* neuronal network," *PLoS Computational Biology*, vol. 7, no. 2, 2011.
- [17] A. T. Winfree, "Biological rhythms and the behavior of populations of coupled oscillators," *Journal of Theoretical Biology*, vol. 16, no. 1, pp. 15–42, 1967.
- [18] G. Jellison Jr and F. Modine, "Parameterization of the optical functions of amorphous materials in the interband region," *Applied Physics Letters*, vol. 69, no. 3, pp. 371–373, 1996.
- [19] A. Mauroy, P. Sacré, and R. Sepulchre, "Kick synchronization versus diffusive synchronization," *IEEE 51st Annual Conference on Decision and Control (CDC)*, pp. 7171–7183, 2012.
- [20] M. Shinriki, M. Yamamoto, and S. Mori, "Multimode oscillations in a modified van der pol oscillator containing a positive nonlinear conductance," *Proceedings of the IEEE*, vol. 69, no. 3, pp. 394–395, 1981.
- [21] R. Miroddo and S. Strogatz, "Synchronization of pulse-coupled biological oscillators," *SIAM Journal on Applied Mathematics*, pp. 1645–1662, 1990.
- [22] Y. Kuramoto, "Self-entrainment of a population of coupled non-linear oscillators," *International Symposium on Mathematical Problems in Theoretical Physics*, pp. 420–422, 1975.
- [23] P. Monzón and F. Paganini, "Global considerations on the Kuramoto model of sinusoidally coupled oscillators," *Proceedings of the 44th IEEE Conference on Decision and Control and European Control Conference (CDC-ECC)*, pp. 3923–3928, 2005.
- [24] F. Dörfler, M. Chertkov, and F. Bullo, "Synchronization in complex oscillator networks and smart grids," *Proceedings of the National Academy of Sciences*, vol. 110, no. 6, pp. 2005–2010, 2013.
- [25] D. A. Wiley, S. H. Strogatz, and M. Girvan, "The size of the sync basin," *Chaos: An Interdisciplinary Journal of Nonlinear Science*, vol. 16, 2006.
- [26] O. V. Popovych and P. A. Tass, "Macroscopic entrainment of periodically forced oscillatory ensembles," *Progress in Biophysics and Molecular Biology*, vol. 105, no. 1, pp. 98–108, 2011.
- [27] M. Porfiri and M. Di Bernardo, "Criteria for global pinning-controllability of complex networks," *Automatica*, vol. 44, no. 12, pp. 3100–3106, 2008.
- [28] J. Lu, D. W. Ho, and Z. Wang, "Pinning stabilization of linearly coupled stochastic neural networks via minimum number of controllers," *IEEE Transactions on Neural Networks*, vol. 20, no. 10, pp. 1617–1629, 2009.
- [29] F. Sorrentino, M. di Bernardo, F. Garofalo, and G. Chen, "Controllability of complex networks via pinning," *Physical Review E*, vol. 75, no. 4, p. 046103, 2007.
- [30] W. Yu, G. Chen, J. Lu, and J. Kurths, "Synchronization via pinning control on general complex networks," *SIAM Journal on Control and Optimization*, vol. 51, no. 2, pp. 1395–1416, 2013.
- [31] Y. Liu, J. Slotine, and A. Barabási, "Controllability of complex networks," *Nature*, vol. 473, no. 7346, pp. 167–173, 2011.
- [32] J. Ruths and D. Ruths, "Control profiles of complex networks," *Science*, vol. 343, pp. 1373–1376, 2014.
- [33] N. E. Leonard, T. Shen, B. Nabet, L. Scardovi, I. D. Couzin, and S. A. Levin, "Decision versus compromise for animal groups in motion," *Proceedings of the National Academy of Sciences*, vol. 109, no. 1, pp. 227–232, 2012.
- [34] D. C. Michaels, E. P. Matyas, and J. Jalife, "Mechanisms of sinoatrial pacemaker synchronization: a new hypothesis," *Circulation Research*, vol. 61, no. 5, pp. 704–714, 1987.
- [35] R. Baldoni, A. Corsaro, L. Querzoni, S. Scipioni, and S. T. Piergiovanni, "Coupling-based internal clock synchronization for large-scale dynamic distributed systems," *IEEE Transactions on Parallel and Distributed Systems*, vol. 21, no. 5, pp. 607–619, 2010.
- [36] G. Filatrella, A. H. Nielsen, and N. F. Pedersen, "Analysis of a power grid using a Kuramoto-like model," *The European Physical Journal B-Condensed Matter and Complex Systems*, vol. 61, no. 4, pp. 485–491, 2008.
- [37] L. Moreau, "Stability of continuous-time distributed consensus algorithms," *Proceedings of the 43rd IEEE Conference on Decision and Control (CDC)*, vol. 4, pp. 3998–4003, 2004.
- [38] P. G. Doyle and J. L. Snell, "Random walks and electric networks," *Carus mathematical monographs*, vol. 22, 2000.
- [39] A. R. Bergen, *Power Systems Analysis*. Pearson Education India, 2009.
- [40] M. Cassidy, P. Mazzone, A. Oliviero, A. Insola, P. Tonali, V. Di Lazzaro, and P. Brown, "Movement-related changes in synchronization in the human basal ganglia," *Brain*, vol. 125, no. 6, pp. 1235–1246, 2002.